Received: 28 April 2023 | Revised: 18 July 2023 | Accepted: 26 July 2023

DOI: 10.1111/tgis.13095

RESEARCH ARTICLE

Transactions in GIS WILEY

DeepIndoorCrowd: Predicting crowd flow in indoor shopping malls with an interpretable transformer network

Chen Chu[1,2] | Hengcai Zhang[1,2] | Peixiao Wang[1] | Feng Lu[1,2]

[1]State Key Laboratory of Resources and Environmental Information System, Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing, China

[2]College of Resources and Environment, University of Chinese Academy of Sciences, Beijing, China

Correspondence
Hengcai Zhang, State Key Laboratory of Resources and Environmental Information System, Institute of Geographic Sciences and Natural Resources Research, Chinese Academy of Sciences, Beijing 100101, China.
Email: zhanghc@lreis.ac.cn

Funding information
National Key Research and Development Program of China [grant number 2021YFB3900803]

Abstract

Accurate and interpretable prediction of crowd flow would benefit business management and public security. The existing studies are challenged to adapt to the indoor environment due to its complex and dynamic spatial interaction patterns. In this study, we propose a crowd flow predicting method for indoor shopping malls, which simultaneously features temporal variables and semantic factors to suit the shopping mall environment. A deep learning model named DeepIndoorCrowd is presented. The model aims at capturing temporal dependencies and the semantic pattern in crowd flow to generate an accurate multi-horizon prediction. With a multi-term temporal dependency capturing structure, the model is effective in learning both daily and weekly patterns of the indoor crowd flow in a shopping mall and is able to provide the temporal interpretation of the prediction result. Moreover, a semantic-temporal fusion module is introduced to utilize the semantic information of stores in prediction, which has proved to be effective in enhancing the model's ability to learn temporal patterns. Experiments were conducted on a real-world dataset to verify the proposed approach. The ablation study demonstrates that the DeepIndoorCrowd can effectively improve the efficiency and accuracy of the prediction up to 18.7%. In addition, some interesting indoor crowd flow patterns were discovered by analyzing the model's interpretation of the prediction result. The

proposed prediction method provides an intuitive way of modeling indoor crowd flow, and the experiment's outcome can help indoor managers better understand stores' flow traffic.

1 | INTRODUCTION

Crowd flow prediction is of great importance to city management (Zheng et al., 2014). Many fatal stampedes have occurred in public areas with high population densities, such as malls and tourist attractions. For example, the Seoul Halloween crowd crush in October 2022 led to 153 fatalities during Halloween celebrations when crowds surged through a narrow alleyway. It is even worse in indoor environments. Recently, the boosting of indoor human mobility datasets has made it possible to monitor people's mobility and predict future positions in any circumstance, with the rapid development of indoor positioning techniques, such as Wi-Fi (Hosseini et al., 2021; Liu et al., 2020), and Bluetooth (Wang et al., 2013). Indoor crowd prediction has attracted many concentrations among all applications due to its high commercial value and variety of applications, including managing special events and enhancing public safety (Zhang et al., 2023).

Although plenty of research has been carried out based on indoor location data and prediction models, including detecting high-density crowd regions (Georgievska et al., 2019; Wang, Gao, et al., 2020) and modeling indoor human mobility (Liu et al., 2022; Trivedi et al., 2021), indoor crowd flow prediction is still challenging because of the following characteristics of the indoor environment. (1) The crowd flow fluctuates greatly during indoor places' opening and closing hours. Most indoor places are managed manually, and their crowd flow variations are extremely sharp. (2) The crowd's transfer pattern varies significantly at different times of the day. The transition flow between indoor places during mealtime and the afternoon could be reversed. It requires both long-term patterns and short-term adjustments to achieve timely prediction. (3) Complex obstacles and walls in indoor environments make the environment hard to be divided into regular partitions. This restriction limits the application of many mature prediction models.

In addition, semantic information is often neglected by existing crowd flow prediction models. Unlike city-wide prediction, indoor space is artificially partitioned, and each location is used for a single purpose. The daily customer flow of an indoor shopping mall is made up of customers that are different groups of people attracted by the same attribute of the store. Hence, factors like the type of business, location, average consumer pricing, etc., determine an indoor store's crowd flow pattern. Semantic information is the most important factor in crowd flow prediction and pattern extraction (Wang, Wang, et al., 2019; Wang, Wu, et al., 2019; Zhu et al., 2021). To illustrate the effect of semantic features, an example is given to show how semantic information affects crowd flow. We normalize the daily crowd flow of each store to a 0 to 1 range. Then we apply the K-Means clustering algorithm to group all stores into two clusters. The average series of each cluster is shown in Figure 1. There are two different crowd flow patterns in all stores. Combined with the store's floor and the store's serving type, in crowd flow pattern 1, about 80% of stores serve as a restaurant. The emergence of the double peaks pattern can be interpreted as the assembling of customers during meal time. This connection between the store type and crowd flow peaks proves the importance of semantic attribute features in flow prediction and the similarity in flow variation serves as prior knowledge of flow patterns, which assists the model in predicting crowd flows in stores of the same type.

In this research, we propose a deep learning model named DeepIndoorCrowd to achieve an accurate and interpretable prediction of indoor crowd flow. With a semantic and temporal features fusion module (STF), the DeepIndoorCrowd introduces the semantic information of indoor stores into prediction. The ablation study proves

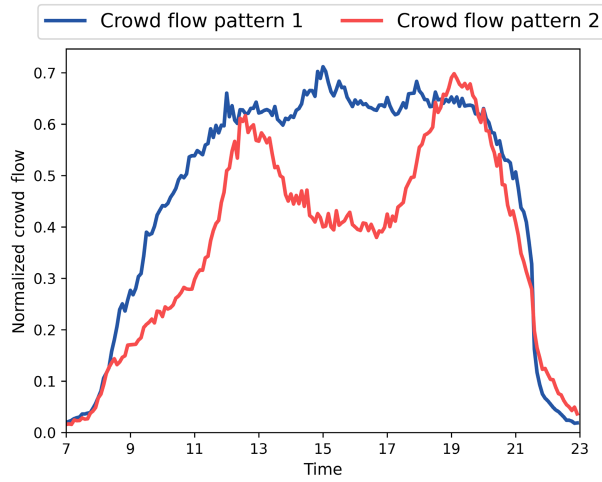

FIGURE 1 The average normalized crowd flow in each cluster when the number of K-Means cluster is set as 2.

that this enhances the model's ability in learning the crowd flow pattern. Additionally, the interpretation result for prediction demonstrates that DeepIndoorCrowd's capacity to capture long-range temporal connections helps the model understand the weekly pattern of crowd flow. We make the following contributions in this article:

- We propose a multi-horizon prediction deep learning model named DeepIndoorCrowd. It is designed to capture temporal dependencies in different temporal scales to suit the complex indoor crowd flow pattern. And a semantic-temporal fusion model is proposed to add semantic information to prediction. The ablation study proves the effectiveness of each part of the DeepIndoorCrowd.
- We model the influence factors of indoor crowd flow from a view of historical temporal, future temporal and semantic features. The structure guarantees the model is able to predict the crowd flows in nonstationary scenes like a weekend or special events. By introducing semantic features into the prediction, the model is able to learn the crowd flow pattern more efficiently.
- We conduct our experiment in a real-world indoor dataset. The results verify the outstanding performance of our model compared with other time series prediction models. The comparison and interpretation also reveal the population dynamic pattern in the indoor environment. By analyzing the interpretation result, the model provides new insights into the indoor crowd flow's temporal pattern.

The remainder of the article is structured as follows. Section 2 introduces the related works. Section 3 defines the indoor crowd flow prediction problem. In Section 4, we construct our DeepIndoorCrowd model and illustrate how it works. Section 5 presents the experimental results with an ablation study and interpretation analysis. Section 6 concludes our work.

2 | RELATED WORKS

The crowd flow prediction at the city scale has been widely studied. The prediction methods can be classified into three types according to the model's structure: (1) grid; (2) graph; and (3) time series based. Grid based models partition the research area into geographical grids. The crowd flow series can be recorded as the channels of these grids. The ST-ResNet (Zhang et al., 2017) is a typical grid based crowd flow prediction method.

After partitioning the city into a grid map. It employs a convolutional network to capture spatial dependencies between nearby and distant regions in a city. Similar methods include SeqST-GAN employs a generative network to make the prediction (Wang, Cao, et al., 2020), and ST-3DNet which employs a three-dimensional convolutional neural network to capture spatial and temporal dependencies at the same time (Guo et al., 2019). Most mature computer vision models can be applied easily thanks to the grid data structure. While it also restricts the method's application in irregular spaces, complex obstacles and walls in an indoor environment restrict the effect of this type of model.

Different from griding the whole region, graph-based methods only model the connection between the regions, so these models are effective in predicting the crowd flow in irregular regions, such as road segments or points of interest. Typical models like Graph Multi-attention Network (GMAN), MVGCN, Diffusion Convolutional Recurrent Neural Network (DCRNN) use transition flow between regions to construct the graph edges, then employ Graph Neural Networks (GNNs) to achieve prediction (Li et al., 2017; Sun et al., 2022; Wang et al., 2023; Zheng et al., 2020). CrowdTelescope proposed a Wi-Fi positioning based multi-grained crowd flow prediction framework for GNN (Zhang et al., 2023). As one of the most common methods to capture spatial–temporal dependencies, GNN aims at modeling the diffusion process between nodes (Cai et al., 2020; Cui et al., 2020; Li et al., 2017; Wang et al., 2022; Zhao et al., 2019). The transfer pattern between nodes must be stable to establish a graph model, and an adjacent matrix using the existing information is essential to represent the pattern. However, unlike the outdoor Euclidean space or road network, indoor space has its unique topology connections formed by complex movement restrictions and dynamic crowd transfer patterns (Lee, 2004; Li et al., 2018; Worboys, 2011), which means it is nearly impossible to model the interaction between indoor nodes by building an invariant adjacent matrix manually.

Time series based methods mainly aim at modeling the crowd flow's temporal pattern of every single region and introducing the additional correlated features to achieve nonstationary prediction. Typical time series forecasting methods include ST-KNN (Cheng et al., 2018) and the Autoregressive Integrated Moving Average model (ARIMA) (Ahmed & Cook, 1979). The Recurrent Neural Network (RNN) is generally used in the deep learning models of the time series based method (Singh et al., 2020). Most contributions focus on modifying the structure of RNN (Li et al., 2020) or using novel generative methods (Rasul et al., 2021). Because the time series prediction of a region does not rely on other regions, the method is more flexible than graph based ones and can generate a more accurate prediction. In recent years, time series based methods are mainly built on sequence-to-sequence structure to achieve multi-horizon forecasts (Cui et al., 2020; Sutskever et al., 2014). However, the traditional Long Short-Term Memory network (LSTM) based structures face problems due to their inability to handle long-range temporal dependencies. Considering the indoor crowd flow has a significant weekly pattern, a more efficient structure is needed. As a model proposed to deal with sequential structures based on an attention mechanism, transformer is effective in capturing long-term reliance in a sequence (Lim et al., 2021; Vaswani et al., 2017; Zhou et al., 2021), and it has been proven to be useful in traffic prediction (Cai et al., 2020) and many other applications (Lim et al., 2021). Furthermore, the attention weights it generates to make a forecast can be utilized to interpret the prediction result, which is useful to reveal the indoor crowd flow pattern and make the prediction reliable.

3 | PROBLEM DEFINITION

3.1 | Preliminary

In this section, we define the basic concept of DeepIndoorCrowd. The schematic diagram of the DeepIndoorCrowd structure is shown in Figure 2.

As shown in Figure 2, to predict the crowd flows accurately, variables that cause or are correlated with the variant of crowd flow need to be modeled properly. We divide related variables into three groups, including historical

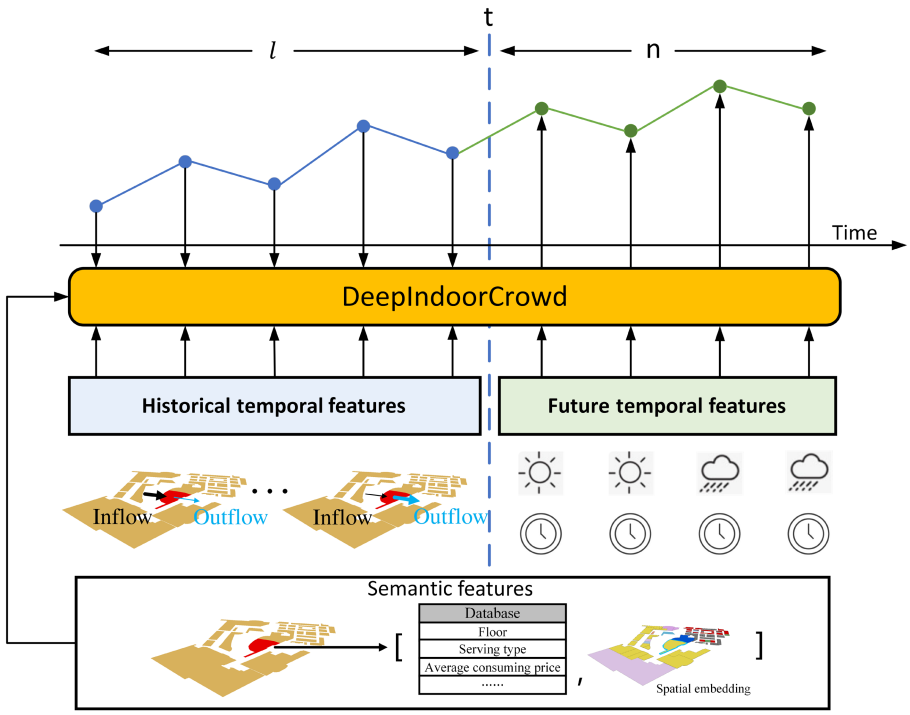

FIGURE 2 Schematic diagram of DeepIndoorCrowd's structure.

temporal features, future temporal features and semantic features. DeepIndoorCrowd is proposed to capture the correlation between indoor crowd flow with these features. It takes a series of historical temporal features, store's semantic features and another series of future temporal features as inputs. Then it predicts a series of the crowd flow for future time stamps. The model can be formulated as the following function:

$$\widetilde{P}^{s}_{(t+1,t+n)}, A^{s}_{(t-l,t-1)} = DeepIndoorFlow\left(HT^{s}_{(t-l,t)}, FT^{s}_{(t+1,t+n)}, Sem^{s}\right) \tag{1}$$

where $\widetilde{P}^{s}_{(t+1,t+n)}$ denotes the predicted crowd flow from time $t+1$ to $t+n$, and n is the length of prediction. $A^{s}_{(t-l,t)} \in \mathbb{R}^{l\times l}$ denotes the attention weight matrix, which records the importance of each time stamp to others, and l is the length of the input historical sequence. $HT^{s}_{(t-l,t)}$ denotes the historical temporal features from time $t-l$ to t. $FT^{s}_{(t+1,t+n)}$ denotes the future temporal features from time $t+1$ to $t+n$. $HT^{s}_{(t-l,t)}$ and $FT^{s}_{(t+1,t+n)}$ are formed using the equations below:

$$HT^{s}_{(t-l,t)} = \left[ht^{s}_{t-l}, ht^{s}_{t-l+1}, \ \dots \ \dots, ht^{s}_{t}\right] \tag{2}$$

$$FT^{s}_{(t+1,t+n)} = \left[ft^{s}_{t+1}, ft^{s}_{t+2}, \ \dots \ \dots, ft^{s}_{t+n}\right] \tag{3}$$

where $ht^{s}_{t} \in \mathbb{R}^{1\times n_1}$ and $ft^{s}_{t} \in \mathbb{R}^{1\times n_2}$, respectively represent the historical temporal features and the future temporal features of store s at time stamp t. n_1 and n_2 respectively denote the number of historical temporal features and number of future temporal features, and each of them represents a variety of features of the moment t. Sem^{s} represents the semantic feature of store s, which is formed by the attribute features and spatial feature of the store:

$$Sem^{s} = \left[\alpha^{s}, Spatial^{s}\right] \tag{4}$$

where Sem^s is formed by the concatenation of attribute feature α^s and spatial feature $Spatial^s$ of store s. The composition of temporal and semantic features vary with the indoor environment. We will later introduce specific temporal and semantic features we modeled in this study in detail to demonstrate how the structure works.

3.2 | Temporal features

3.2.1 | Historical temporal features

Variables that change throughout time can provide direct information for the variation of crowd flow. Daily, weekly, or even annual temporal patterns of crowd flows are learned from these temporal dynamic features. Modeling the temporal features aims at featuring the periodicity of crowd flow and its covariance with other factors. The specific composition of historical temporal features modeled in DeepIndoorCrowd is denoted as

$$ht_t^s = \left[N_t^s, I_t^s, O_t^s, T_t, \varphi\left\{t_{week}, t_{day}, t_{hour}, t_{minute}\right\}\right] \tag{5}$$

where N_t^s denotes the number of customers in store s at timestamp t, I_t^s, and O_t^s, respectively denote the number of customers who entered and the number of customers who left store s at time stamp t. T_t denotes the total number of customers in the indoor shopping mall at time stamp t. φ represents a temporal embedding function. The function first converts the current time recorded by $t_{week}, t_{day}, t_{hour}, t_{minute}$ into timestamps, such as the month of the year, time of the day, or day of the week. Then these timestamps are converted into a trainable temporal embedding. In this study, we only use the time-of-day and day-of-week. We first encode the timestamp with one-hot coding. Then a trainable, fully connected neural network is employed to map the discrete-time encoding into a continuous feature space, which allows the model to learn the temporal relationship in crowd flow variation as follows:

$$\varphi\{t\} = concate\left[Encoder_{onehot}\left(t_{day-week}\right) * W_{day}, \ \ldots, Encoder_{onehot}\left(t_{time-day}\right) * W_{time}\right] \tag{6}$$

where $Encoder_{onehot}$ is a one-hot sorting function, $W_{day} \in \mathbb{R}^{day \times h}$, and $W_{time} \in \mathbb{R}^{time \times h}$ are the trainable weights of time embedding function, *day* and *time* are the number of the day-of-week and time-of-day, and h is the number of hidden dimensions for the temporal embedding.

The variant of the number of customers N_t^s in each store follows a temporal pattern and the autocorrelation of crowd flow series is the most fundamental factor for prediction. It is reflected by the stationary daily and weekly patterns. Its temporal dependence directly contributes to the model's ability to predict the crowd flow of regular moments. Therefore, learning the variation tendency of the flow series is a preliminary task for the model.

We form a real-time crowd flow transition matrix M_{ij}^t by accumulating the number of people transferred from store to store in each time slice. Every column i and row j represent a store in the market. M_{ij}^t represents the number of people transferring from store i to store j in timestamp t. Two transition matrices from different times of the same day are shown in Figure 3.

In Figure 3, compared with the transition matrix at lunchtime, the transition frequency between stores rises significantly in the afternoon. Transition flows with high frequency also vary with time. Moreover, a large proportion of elements only record a value of zero, which means the transition matrix contains redundant information. To simplify the calculation and reduce the inference time, we use the indegree I and outdegree O to represent the interaction flow between stores, which are calculated as follows:

$$I_t^s = \sum_{j=0}^{N} M_{sj}^t \tag{7}$$

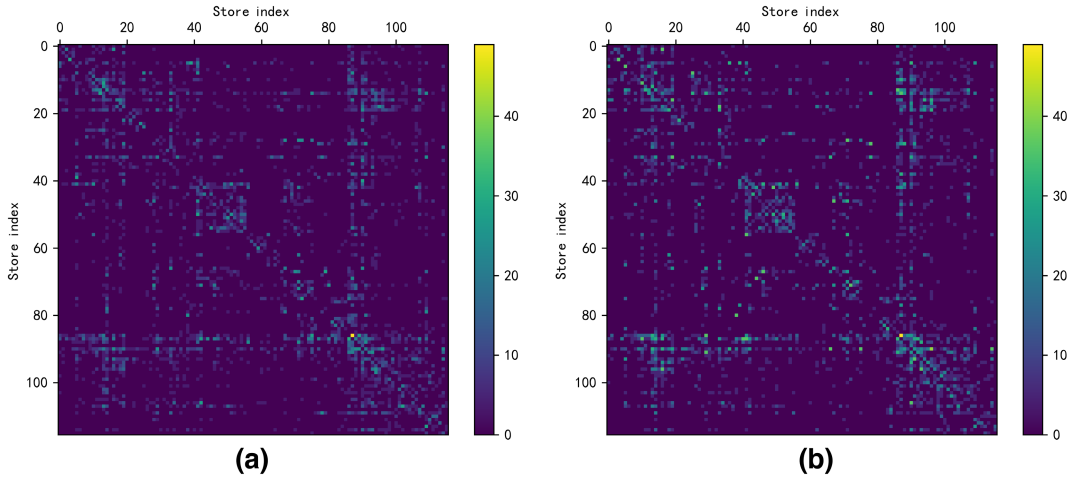

FIGURE 3 (a) Crowd flow transition matrix between 12:00 and 12:05 on a regular weekday; and (b) transition matrix during 15:30–15:35 on the same day.

$$O_t^s = \sum_{i=0}^{N} M_{is}^t \tag{8}$$

where I_t^s, O_t^s, respectively, denote the number of people who entered and exited the store s at time t.

These two features are the driving factors in the variation of the crowd flow series. Therefore, indegree and outdegree features benefit the model in predicting anomaly fluctuation caused by occasional events. For instance, when a sale promotion is held in a particular store, the abnormal increment of the indegree I will happen ahead of the rise of the crowd flow. These kinds of correlations captured in the training dataset will increase the accuracy of the prediction in abnormal circumstances.

Based on the assumption that the number of customers who entered a specific store is strongly correlated with the number of customers who entered the building. We modeled the number of customers who entered the indoor environment T_t. The days when festivals are held are always rare. The nonstationary variation makes crowd flow variants in these days hard to predict. Considering this feature provides the trend of crowd flow variance, it also benefits the model with anomaly prediction. An example of the total crowd flow and the crowd flow of a selected store is shown in Figure 4.

As shown in Figure 4, different from weekdays, on weekends, the total crowd flow is at its highest point at midday rather than in the evening. This circumstance affects the crowd flow of the plotted store. Obviously, the crowd flow of the store plotted in light color follows the pattern. The correlation shows the effect of feature T_t. Furthermore, compared with the outdoor environment, indoor environments like shopping malls usually are managed manually, which means, regularly, the crowd flow volume in indoor environments may vary dramatically in moments like the opening and closing period of the store. The feature can also provide the opening or closing information to the model.

3.2.2 | Future temporal features

Future temporal features are taken as the input of the decoder. It aims at guiding the model to deal with expected events. As shown in Figure 2, future temporal features model events we can foresee previously. For instance,

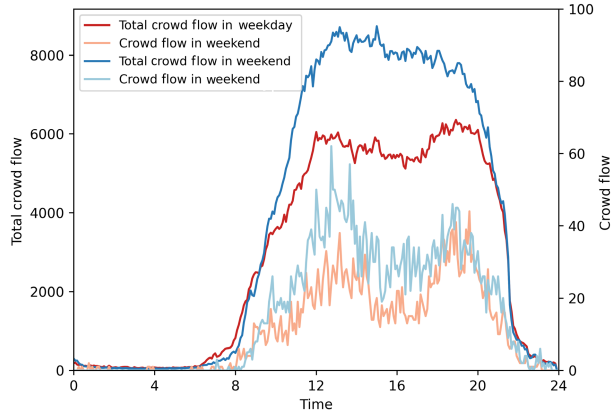

FIGURE 4 Comparison of crowd flow on weekdays and weekends.

the sales promotion in a store is arranged by its manager, so its arrangement can be found in the schedule ahead of time. Future features provide information to crowd flow variation in these moments so that the DeepIndoorCrowd model can generate predictions based on the real-time situation. These kinds of features are named future temporal features, and they can be formulated as follows:

$$ft_t^s = \left[sell_t^s, holiday_t, \varepsilon\left(w_t\right), \varphi\left\{t_{week}, t_{day}, t_{hour}, t_{minute}\right\}\right] \tag{9}$$

$$\varepsilon\left(w_t\right) = Encoder_{onehot}\left(w_t\right) * W_{weather} \tag{10}$$

where $sell_t^s$ represents the sale promotion of store s at time t. It is set as a continuous variable representing the promotion's strength. $holiday_t$ represents the scheduled holiday at time t, it is presented as a one-hot vector. The same temporal embedding function φ is employed to capture temporal dependency. Additionally, the weather for the upcoming few hours predicted by the weather prediction may also be taken as a given. We embed different types of weather w_t at time t with function ε. The weather embedding function also uses a fully connected neural network to map the one-hot representation of different weather types into a feature space. $W_{weather} \in \mathbb{R}^{weather \times h}$ are the trainable weights of weather embedding function, *weather* is the type of weather embedded, h is the number of hidden dimensions of the weather embedding. Even in special cases, like holidays or sales promotions, these features can still promise the prediction of the DeepIndoorCrowd.

3.3 | Semantic features

In a city-scaled crowd flow prediction model, the temporal crowd flow pattern is formed by the movement activity of the city's residents, and their movement pattern is fixed during a period of time. While different from that, the customer flow of an indoor store is made up of the consumers who visited the store each day, which are different groups of people attracted by the same attributes of the store. Therefore, the temporal crowd flow pattern of an indoor store is decided by its attributes like type of store, geographic location, average consuming price, etc. Attributes like these are extracted as attribute features. The attribute features modeled by DeepIndoorCrowd are listed in Table 1.

The attribution information was collected from the online customer serving platform and the map of the indoor mall. For those categorical features, we employed the feature embedding method to map them into continuous vector space as follows:

$$SE(x) = Encoder_{onehot}(x) * W_x \tag{11}$$

TABLE 1 Indoor stores' semantic features.

Name of feature	Embedding size
Floor	2
Serving type	2
Occupied area	1
Average consuming price	1
Identification embedding	8

$$\alpha^s = concate\left[SE_1\left(x_1^s\right), \ \ldots \ \ldots, SE_n\left(x_n^s\right)\right] \tag{12}$$

In Equation (11), $Encoder_{onehot}$ is a one-hot sorting function, which maps the input category x into a one-hot representation vector and $W_x \in \mathbb{R}^{l \times h}$ is the trainable weights of feature x embedding function SE. l and h, respectively, represent the number of categories of feature x and the dimension of the embedded vector. α^s is formed by concatenating the embedding feature of all the above attributions. With the help of the feature embedding, DeepIndoorCrowd is able to learn the relationship between discrete variables.

3.4 | Spatial features

We employ the pretrained spatial feature to represent stores' spatial relationships. Considering customers' shopping behavior is directly affected by stores' spatial locations, customers' trajectories can reflect the spatial relationship between stores. Therefore, we employ the Global Vectors for Word Representation (GloVe) algorithm (Pennington et al., 2014) to extract the spatial embedding vector of each store from historical crowd trajectories. We used the crowd flow transition matrix M^t that had been created earlier when we extracted historical temporal features. By summing up the transition matrix of all moments, we got the overall transition matrix from store to store. Then we pretrained the spatial embedding vector as follows:

$$X_{ij} = \sum_t M^t \tag{13}$$

$$\text{loss} = \sum_{i,j=1}^{N} f\left(X_{ij}\right)\left(Spatial^{s_i T} \bullet Spatial^{s_j} - \log X_{ij}\right)^2 \tag{14}$$

$$f\left(X_{ij}\right) = \begin{cases} \left(X_{ij}/X_{\max}\right)^{0.75} & \text{if } X_{ij} < X_{\max} \\ 1 & \text{otherwise} \end{cases} \tag{15}$$

where $Spatial^{s_i} \in \mathbb{R}^d$ is the spatial embedding feature of store s_i, d is the dimension of spatial embedding, $X_{ij} \in \mathbb{R}^{N \times N}$ is the of cooccurrence frequency of store i and store j, and N is the number of stores. By randomly picking up two stores i, j and minimizing the loss function $loss$, GloVe leverages the statistical information from the cooccurrence between two stores. Through several iterations, it produces a vector space that can embed these stores by their transition relationship. The spatial feature embedded by the GloVe algorithm makes the model able to determine the relative spatial location of a store and learn its crowd flow pattern from transfer relations. These features are frozen when training the prediction model. In this study, we set $d = 8$ as the dimension of the spatial feature of each store.

4 | DEEPINDOORCROWD

4.1 | DeepIndoorCrowd architecture

In order to achieve the interpretable prediction of indoor crowd flow, we design a sequence-to-sequence crowd flow prediction deep learning network named DeepIndoorCrowd, which integrates the transformer network, the LSTM network and a feature fusion module STF. The key contributions of the model are its focus on multi-term temporal dependencies capturing and semantic-temporal features fusion. The structure of our DeepIndoorCrowd model is shown in Figure 5.

As shown in Figure 5, the DeepIndoorCrowd follows the encoder-decoder structure. The encoder LSTM and transformer network collect information from historical temporal features and embed the information into a feature space. For the transformer encoder, the encoded information is memorized in a sequence of continuous representation, which is the output memory of the transformer encoder block and it is used as the input of the transformer decoder. For the LSTM structure, the decoder LSTM network learns previous information by initializing its hidden state and cell state with the last state of the encoder LSTM network. One of the problems the transformer faces in time series prediction is that the model does not have the concept of order, which means that to a particular time stamp, inputs in its nearby position play the same role as inputs far away. In DeepIndoorCrowd, the ignorance of relative position in the input sequence can be filled up by the LSTM layer we employed before transformer. We will further introduce each part of the model in the remainder of this section.

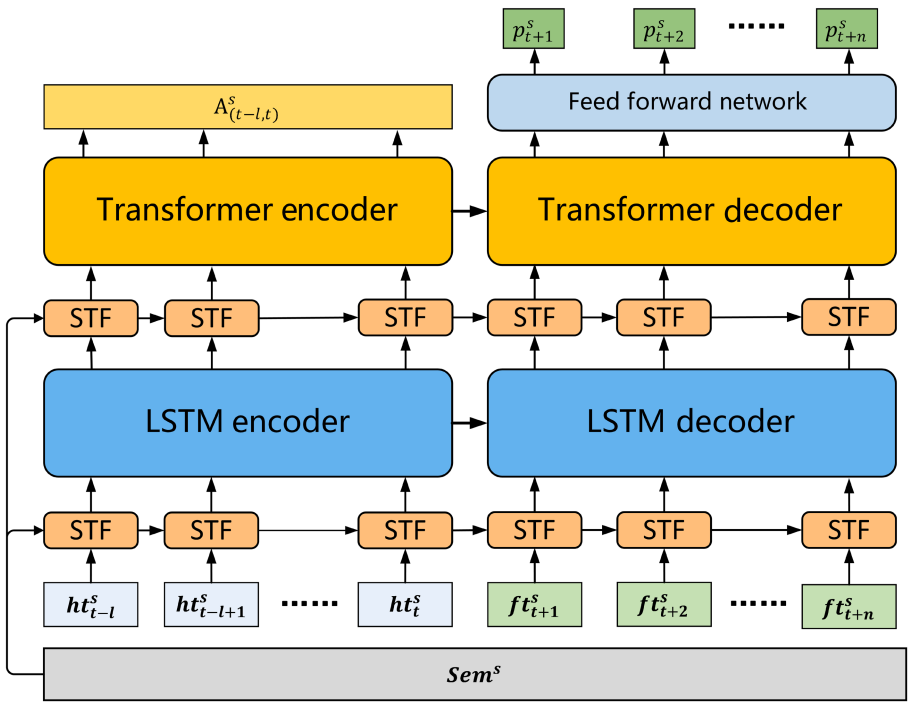

FIGURE 5 Architecture of DeepIndoorCrowd.

4.2 | Semantic-temporal features fusion

We employ the STF as a building block of DeepIndoorCrowd to achieve the fusion of semantic and temporal features. The fusion is expected to enable the model to learn from the relationship between the store's semantic attributions and its crowd flow pattern. The STF is formulated as:

$$\widetilde{d}_t^s = STF\left(d_t^s, Sem^s\right) = Norm\left(d_t^s + GLU(\delta)\right) \tag{16}$$

$$\delta = Concat\left(d_t^s, Sem^s\right) * W_1 + b_1 \tag{17}$$

$$GLU(\delta) = \left(\delta * W_2 + b_2\right) \odot \sigma\left(\delta * W_3 + b_3\right) \tag{18}$$

where $d_t^s \in \mathbb{R}^r$ denotes the input temporal feature at time t, r is number of dimensions for the temporal feature. $Sem^s \in \mathbb{R}^m$ denotes the semantic feature of the store s, m is number of dimensions for the semantic feature. $W_1 \in \mathbb{R}^{(r+m)\times h}$, $W_2 \in \mathbb{R}^{h\times r}$, $W_3 \in \mathbb{R}^{h\times r}$ and $b_1 \in \mathbb{R}^h$, $b_2 \in \mathbb{R}^r$, $b_3 \in \mathbb{R}^r$ are the trainable weights and biases of the STF block. h is the number of the number of hidden units in STF. *Norm* is standard layer normalization operation (Ba et al., 2016). GLU is a Gated Linear Unit used to suppress unrequired architecture (Dauphin et al., 2017), $\odot$ is the element-wise Hadamard product. The illustration of STF's connection is shown in Figure 6.

With a fully connected layer added after the concatenate operation, the module has the flexibility to fuse two types of features in any combination. After that, the gated mechanism is employed in GLU to help the module select useful semantic features to join in the prediction. With a sigmoid function, the GLU's output can be closed to 0, which means the inefficient features can be completely abandoned and the module can control the participation of semantic information. To avoid the disturbance from a part of the useless semantic features, the residual connection in STF can still preserve the original information in temporal features. In the DeepIndoorCrowd, the semantic features are fused twice separately before the LSTM layer and the transformer. This promises that the semantic information will not get lost after the sequential calculation of the LSTM network.

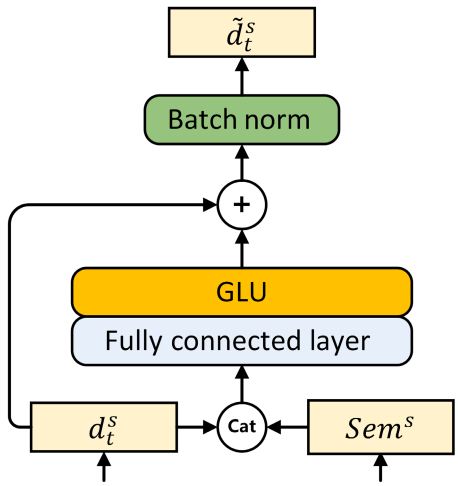

FIGURE 6 Architecture of STF.

4.3 | Capturing multi-term temporal dependencies

Indoor crowd flow has strong temporal patterns in both daily and weekly or monthly scales. While these patterns all contribute to the prediction of the current status, we proposed a multi-term temporal dependencies capturing structure. Firstly, we employ the LSTM network (Hochreiter & Schmidhuber, 1997) to capture the temporal patterns in short-term variation. The LSTM network has been widely used in modeling sequential data. With the gates mechanism and serializing data dealing structure, the LSTM network can capture the temporal dependencies in a relative short term and encode positional information into the series, which aims to provide an appropriate inductive positional encoding method for the attention mechanism of the transformer. The equations of a LSTM unit are as follows:

$$i_t = \sigma\left(W_i * h_{t-1} + U_i d_t^s\right) \tag{19}$$

$$o_t = \sigma\left(W_o * h_{t-1} + U_o d_t^s\right) \tag{20}$$

$$f_t = \sigma\left(W_f * h_{t-1} + U_f d_t^s\right) \tag{21}$$

$$\widetilde{c}_t = tanh\left(W_c * h_{t-1} + U_c d_t^s\right) \tag{22}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c}_t \tag{23}$$

$$h_t = o_t \odot \tanh\left(c_t\right) \tag{24}$$

where d_t^s, h_t, and c_t respectively denote the input temporal feature of store s, the hidden state, and the cell state at time t. h_{t-1} is the hidden state of the layer at time $t-1$ or the initial hidden state at time 0. i_t, o_t, f_t, $\widetilde{c}_t$ are the input, output, forget and cell gates at time t. W and U are the trainable parameters for the corresponding gate and cell. Every output from LSTM's encoder and decoder network is encoded with its relative position in the sequence and the information from the temporal features of its previous timestamps. Then the corresponding output states are passed to the transformer encoder and decoder separately. The transformer's architecture is shown in Figure 7.

As shown in Figure 7, $\widetilde{d}_t^s$ denotes the output of the STF module at time t. We employ the transformer network to capture long-term dependencies in the output sequence of the previous network. As a sequence-to-sequence architecture, its encoder block maps an input sequence of dynamic time series features to a sequence of continuous representations named memory, which contains the information from the input series. Then the transformer decoder block decodes an output series based on the memory. Different from the LSTM network capturing temporal dependencies by mapping inputs in a serialization way, the transformer replies on the scaled dot product attention mechanism to capture the relationship between inputs. The attention mechanism is formulated as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{25}$$

where Q, K, V represents query, key and value, respectively. They are mapped by three separated linear layers from the same input d_t^s. d_k is the dimension of the key. The outputs of the *softmax* function are the attention weights between positions. By calculating the similarity between every two timestamps directly, the attention layer is allowed to pick up long-term dependencies that may be challenging for the LSTM layer to learn. This ability is of significant importance for DeepIndoorCrowd to capture weekly patterns in the indoor crowd flow. Furthermore, the attention function can generate attention weights after the model is trained, which denotes the importance of each position in a sequence. This provides the model with interpretability for its prediction.

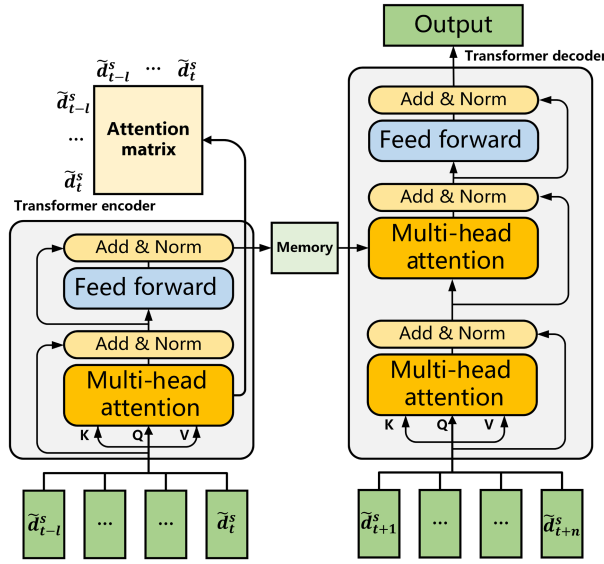

FIGURE 7 Architecture of long-term temporal dependencies capturing transformer.

TABLE 2 Example of indoor positioning dataset.

Time stamp	Floor ID	User ID	X	Y
2017-12-31 08:05:54	F7	341298C7****	13****79.9	4****50.8
2017-12-31 08:06:42	F6	341298C7****	13****62.0	4****66.6
.....	.....	.....	.....	.....
2017-12-31 19:43:35	B1	28FAA07D****	13****43.5	4****08.1

5 | EXPERIMENT

We conduct the experiment on a real-world indoor positioning dataset collected from a shopping mall in China. We compare our model with several other baseline methods. Then by analyzing the attention weights generated by DeepIndoorCrowd, the prediction results are interpreted and crowd flow patterns of the indoor shopping mall are unveiled.

5.1 | Dataset

An indoor wi-fi positioning dataset is used for the experiment. It recorded the positioning result of all customers who visited the shopping mall carrying their mobile phones. The dataset includes positioning data for the entire eight floors of the shopping mall from September 20, 2017 to February 1, 2018. We use the crowd flow series of the previous 37 days to train our model, and test its performance in the last 7 days. A few sample positioning points are shown in Table 2. We hide some parts of the data to browse anonymously and protect users' privacy.

We form the crowd flow data of each store by compiling statistics of the number of user IDs with more than three positioning points located in a store. The statistical division is set as 5 min. Eventually, there are 116 stores in the shopping mall that satisfy the basic requirement for prediction. A brief view of the stores' spatial distribution and their classification is provided in Figure 8.

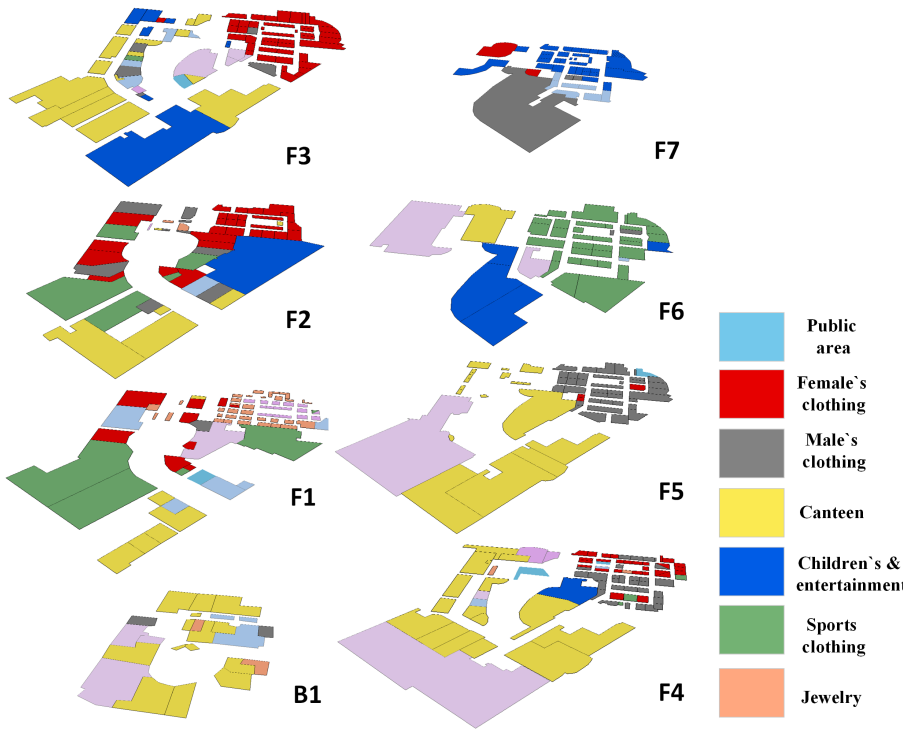

FIGURE 8 Spatial distribution and classification of stores in the mall.

Furthermore, we construct the trajectory of each customer from its positioning points. Through applying the Indoor-STDBSCAN algorithm (Wang, Wang, et al., 2019) to the raw trajectory formed by positioning points, the semantic trajectory that records the customer's visit to each store is extracted. Then the semantic trajectory was used to train the spatial feature $Spatial^s$ of each store.

5.2 | Implementation details

DeepIndoorCrowd is able to deal with sequence data with diverse lengths. We take the sequence of temporal features from 8 days before the predicting moment as input. Then the model generates the prediction of the crowd flow volume for the next 15 or 30 min. The model is trained using the Adam optimizer with the initial learning rate set as 1×10^{-3}. The MSE loss is used as the training loss of our model. The model was trained on a Nvidia Titan V GPU and implemented by Pytorch.

There are some key hyperparameters for the DeepIndoorCrowd model. We used the cross-validation method to find out the hyperparameters that ensures the model generates the most accurate prediction. There are three key hyperparameters: (1) the number of hidden units in LSTM and transformer encoder-decoder; (2) the number of hidden units in the STF module; and (3) the number of heads in the multi-head attention module. In order to weigh the accuracy and the time consumption, the number of heads in the attention module is set as 8, and the number of layers in the encoder and decoder structure of LSTM and transformer are both set as 1. Then we compared the model's performance with different hidden units in the semantic fusion and temporal modules. The result is shown in Figure 9.

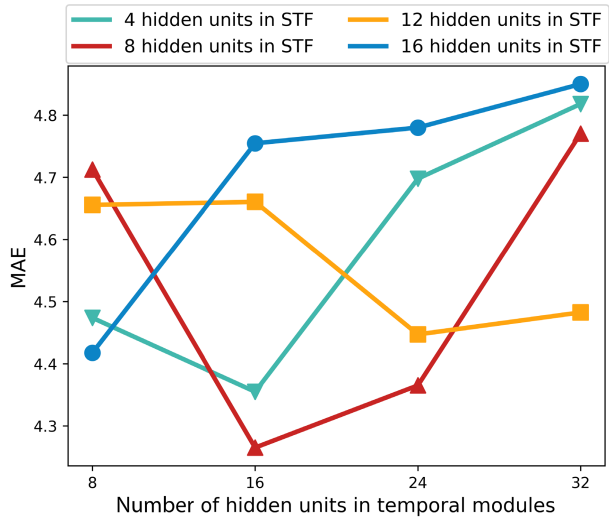

FIGURE 9 Model's performance with different numbers of hidden units.

According to the comparison in Figure 9, with the same training epoch, the best result occurs when the number of hidden units in both the transformer and LSTM modules is 16 and the number of hidden units in STF is 8. The outcomes also show that too many hidden units in the semantic fusion part may have an adverse impact on prediction.

5.3 | Evaluation metrics

As a regression problem, we adopt 3 commonly used metrics to evaluate the performance of different models. The metrics are defined as follows.

Root Mean Squared Error (RMSE),

$$RMSE(X,\overline{X}) = \sqrt{\frac{1}{N}\sum_{i=0}^{N}(X_i - \overline{X}_i)^2} \tag{26}$$

Mean Absolute Error (MAE),

$$MAE(X,\overline{X}) = \frac{1}{N}\sum_{i=0}^{N}\left|X_i - \overline{X}_i\right| \tag{27}$$

Mean Absolute Percentage Error (MAPE),

$$MAPE(X,\overline{X}) = \frac{1}{N}\sum_{i=0}^{N}\left|\frac{X_i - \overline{X}_i}{X_i}\right| \times 100\,\% \tag{28}$$

where N is the number of predicted values, X_i represents the ground truth value of the crowd flow, and $\overline{X}_i$ represents the predicted values. The smaller the three metrics are, the better the prediction result suits the actual situation.

5.4 | Baselines

5.4.1 | Historical average

Historical average (HA) models the crowd flow as a weekly process using the averaged data of previous corresponding moments to predict the current state. In our experiment, we use the averaged crowd flow of the corresponding moment in the same days of two previous weeks as the prediction result. The method outputs the same result in both 15 and 30 min predictions.

5.4.2 | Autoregressive integrated moving average

Autoregressive integrated moving average (ARIMA) is the most classic model for forecasting a time series. By calculating the discrete difference of the sequence and observing the autocorrelation plot and partial autocorrelation plot, we selected the p as 75, q as 6 and d as 1 for prediction.

5.4.3 | Diffusion convolutional recurrent neural network

Proposed by Li in 2017 (Li et al., 2017), the diffusion convolutional recurrent neural network (DCRNN) is one of the benchmark methods for spatial–temporal prediction. It combines a graph convolution neural network with a recurrent neural network and models the diffusion process between nodes in both spatial and temporal domains. After fine-tuning the parameters of the model, we employed a DCRNN model with both decoder and encoder containing 2 recurrent layers and containing 64 hidden units for each layer. The curriculum learning is employed to train the model and the initial learning rate is 1×10^{-2}. The connection graph is constructed based on the transferring frequency between stores and it takes the crowd flow of the previous 3 h to make the prediction.

5.4.4 | Graph multi-attention network

Graph multi-attention network (GMAN) is another graph based spatial–temporal prediction model (Zheng et al., 2020). Based on the graph attention mechanism, the model can capture complex spatial–temporal correlations. We fine-tuned the hyperparameters setting from its original proposal and used the spatial feature we extracted from GloVe as the graph embedding feature. The number of attention heads is set as 8, the output dim of attention heads is set as 8 and the number of attention blocks is 1. The model takes the crowd flow of the previous 1 h as input.

5.4.5 | Fully connected LSTM

Fully connected LSTM (FC-LSTM; Sutskever et al., 2014) is one of the most important baseline models in time the series forecasting research. With a sequence-to-sequence architecture, the model employs two LSTM networks as encoder and decoder and outputs the prediction with a fully connected layer. We feed the model with the same temporal features as the DeepIndoorCrowd takes. The number of hidden units of LSTM is set as 32 after fine-tuning the prediction accuracy.

5.4.6 | Informer

Informer is an improved form of the transformer for the time series forecasting task (Zhou et al., 2021). It is efficient in handling long input sequences and generates long horizons forecasts. We also fine-tuned the hyperparameters setting. The dimension of the hidden layer is set as 64 and the number of heads is set as 8. The number of layers for encoder and decoder is set as 1. It also takes the same input as the DeepIndoorCrowd.

5.5 | Performance evaluation

We compare the prediction accuracy in 15 and 30 min of DeepIndoorCrowd and the other baselines to evaluate their ability at different time scales. Moreover, we also compare the performance on weekdays and weekends. With more people entered into the shopping mall in weekends, the variation of crowd flow becomes harder to predict, and there is also less training data for weekends than that of weekdays. Table 3 shows the prediction performance of different models with RMSE, MAE, and MAPE as evaluation metrics.

There are several phenomena shown in the experiment result. First, the graph based spatial–temporal predicting model DCRNN and GMAN generate the worst prediction results. Even though we have used training methods like curriculum learning and tried different graph connecting methods like Euclidean distance based connection or transfer frequency based connection, they still cannot achieve further improvement. This phenomenon illustrates that indoor stores' topology connection is much more complex than a single graph can represent. It could be dynamic from time to time. Besides, the number of visitors varies greatly from store to store and also makes it hard for a graph-based model to converge to the best performance for every node. During the closed period of the mall, there are many timestamps with the number of their crowd flow is zero. This may make it difficult for the graph based model to learn from the series. Moreover, the graph convolution operation aims to simulate the diffusion process of people between nodes. However, different from the traffic flow constrained by the road network, people in the indoor environment have more options and their selection of the next visiting spot is affected by semantic features. Thus, the uncertainty of human movement in an indoor environment is hard to model appropriately with an integral model.

Second, transformer based models like Informer and ours, tend to have better performance compared with the recurrent structured FC-LSTM network. This is because transformer can capture temporal dependency in the longer term. Indoor crowd flow varies greatly during a short time and the accuracy of prediction depends much more on the long-term tendency captured by the prediction model. By calculating the dot-product between timestamps, the multi-head attention module in transformer can focus on the most relevant inputs in the sequence without the constraint of order. This advantage of the transformer is more apparent, especially when the input sequence is extremely long. However, without the assistance of semantic information, the Informer still cannot achieve the best performance. This may be because the model cannot identify which of the stores the input sequence belongs to, so it is unable to form memories of the crowd flow pattern of a particular store.

Third, our DeepIndoorCrowd achieves the best performance with all metrics, which proves the importance of semantic information in classifying the crowd's dynamic pattern of indoor stores, and the effectiveness of the stores' interaction intensity feature we extract. The semantic fusing structure with add and norm operations preserves the original information of crowd flow series while mixing the sequential embedding with corresponding semantic information that can assist prediction. The outstanding performance of DeepIndoorCrowd can also be attributed to the way it combines the recurrent structure with transformer so that the transformer can calculate attention weights with the awareness of inputs' relative position.

Figure 10 shows the DeepIndoorCrowd's predicted result of the different stores on a Saturday. The graphs show that the model accurately predicted the increase in visitors in the store's opening hour with nearly no delay. Despite the outliers, the predicted peak value of the day is close to the ground truth in the 15 min ahead prediction.

TABLE 3 Performance comparison of different models.

	Weekday						Weekend					
	15 min			30 min			15 min			30 min		
	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)	RMSE	MAE	MAPE (%)
HA	9.1360	6.9410	28.36	9.1360	6.9410	28.36	10.3824	7.7648	26.89	10.3824	7.7648	26.89
ARIMA	10.9973	9.7406	24.72	9.5952	7.9152	32.53	10.7347	9.8354	34.52	11.3624	9.6630	52.72
DCRNN	11.6683	7.0980	28.21	15.2174	8.8655	36.85	16.2058	9.3677	32.92	20.1287	12.0705	41.71
GMAN	11.3697	6.3654	30.96	11.5153	6.4104	31.88	12.8900	8.11987	33.55	13.1485	7.6251	32.46
FC-LSTM	7.9084	5.6518	26.45	8.2221	5.8362	26.99	15.3555	8.4868	20.62	9.7683	7.0488	30.13
Informer	6.2307	5.1124	26.77	6.8149	5.2667	30.53	6.7962	5.6157	29.32	8.2088	6.5877	26.60
DeepIndoorCrowd	5.9383	4.0537	18.15	6.2668	4.2478	21.24	6.1870	4.2763	19.16	6.5876	4.5251	19.75

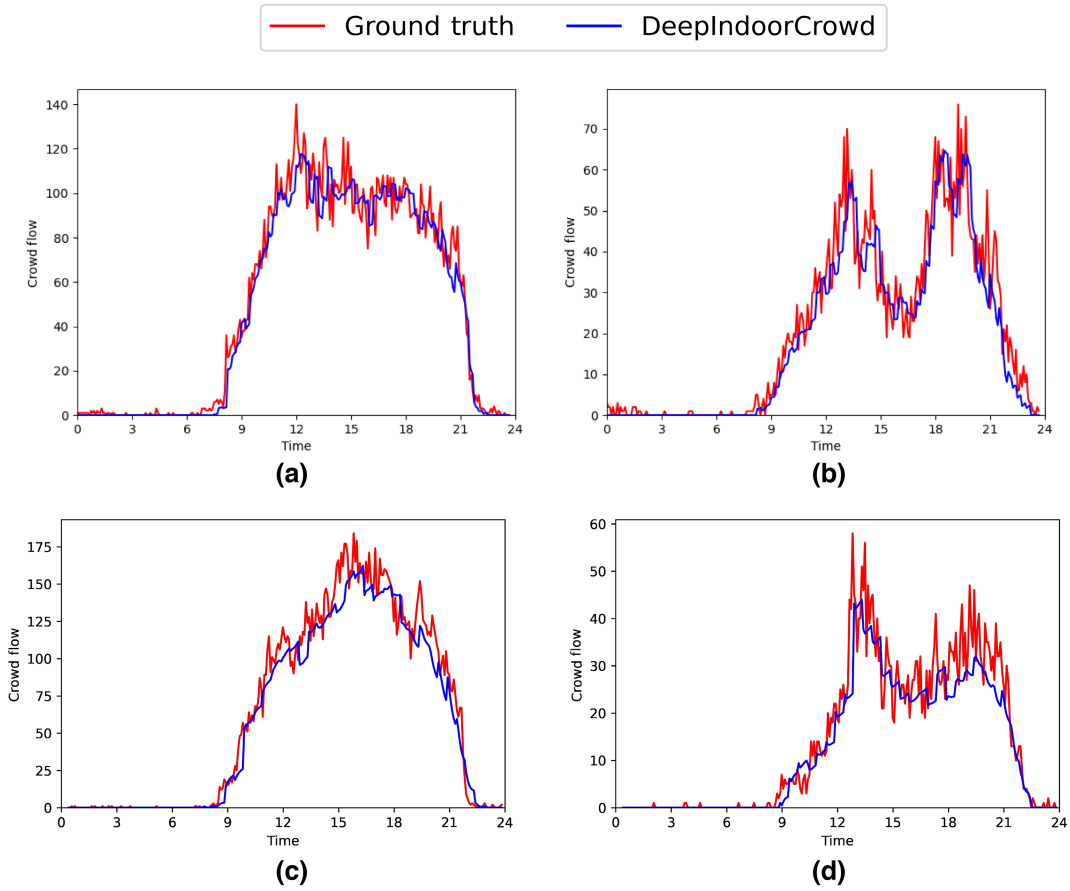

FIGURE 10 Prediction performance of looking 15 min ahead (a, b) and 30 min ahead (c, d).

For the 30 min ahead prediction, the model is still able to predict the variation trend of the crowd flow, but tends to underestimate the crowd flow of the second peak in the evening. The model successfully distinguished two different types of crowd flow and forecast a double peak for the catering store.

5.6 | Ablation study

We perform an ablation study by comparing the performance of different models during the training process. We compare three models as follows: (1) LSTM-STF: the model combines the semantic-temporal fusion module with the LSTM network. All its inputs are the same as with DeepIndoorCrowd. (2) LSTM-transformer: the model inherits the temporal capturing structure of DeepIndoorCrowd but removes the semantic fusion module. The input temporal features are the same as with DeepIndoorCrowd. (3) DeepIndoorCrowd: the model we proposed. We trained these three models with the same initial seed, learning rate, and optimization method. The performance of their RMSE in the evaluation dataset in the first 20 epochs is shown in Figure 11.

Obviously, the DeepIndoorCrowd achieves the best performance after the first few epochs and constantly reduces its evaluation error. It achieves an 18.72% increment in RMSE compared with the best performance of others. There are a few interesting phenomena. The LSTM-transformer model is not able to introduce the store's

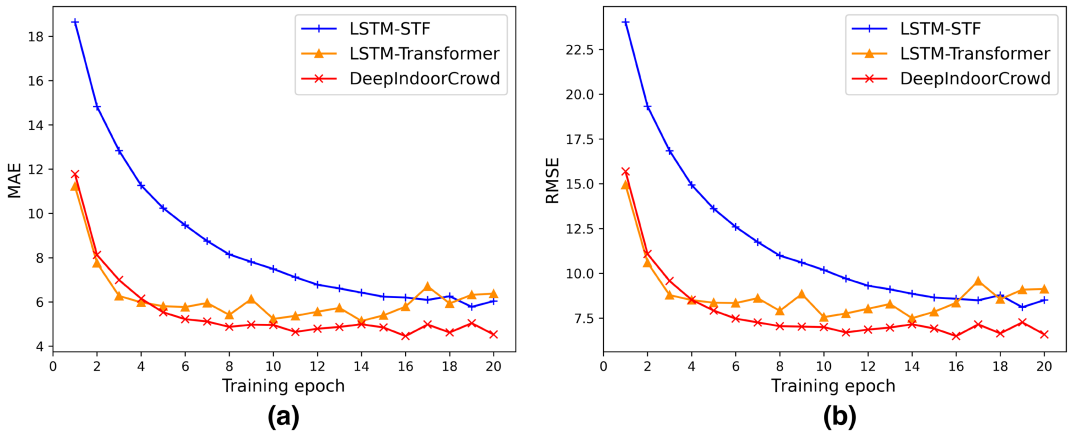

FIGURE 11 Performance comparison of different models during training.

semantic features into prediction. Although its evaluation error is quickly reduced during the first few steps, it cannot achieve better performance and even starts overfitting after that. The reason is that the model is not able to determine which store the input time series sequence belongs to, and cannot summarize the attribute information to learn the crowd flow's temporal pattern for different types of stores. The performance of the LSTM-STF model surpasses the pure temporal model at last. The model lacks the capacity to directly capture long-term temporal dependencies. Therefore, its evaluation error reduces slowly. This proves the importance of weekly patterns in indoor crowd flow prediction.

5.7 | Model interpretation

Attention weights in the transformer model reveal the importance of each input timestamp. To make the prediction result more convincible and more reliable, we visualize the attention weights generated by the encoder of the transformer. The attention weights assigned to each input time stamp from other positions are summed up using the equations as follows:

$$\text{Importance}^{s}_{(t,t-l)} = \left[i^{s}_{t-l}, i^{s}_{t-l+1}, \ \cdots \ \cdots, i^{s}_{t}\right] \tag{29}$$

$$i^{s}_{t} = \frac{\sum_{i=0}^{l} \mathrm{A}^{s}_{(t,t-l)}(t,i)}{l} \tag{30}$$

$\text{Importance}^{s}_{(t,t-l)}$ is the interpretable series, which denotes the importance of each time stamp from $t - l$ to t. Its element i^{s}_{t} is the average attention weight calculated from the mean of column t in the attention weights matrix $\mathrm{A}^{s}_{(t,t-l)}$. The more attention weight one time stamp received, the more information it provided to the decoder's prediction result. Plotting the crowd flow of the input sequence and the averaged attention weight on its corresponding position, Figure 12 shows the attention's temporal pattern $\text{Importance}^{s}_{(t,t-l)}$ of several typical stores while predicting the 17:00 crowd flow of a Saturday in the validation dataset. By analyzing the distribution of attention weights, we can interpretate the predicting mechanism of the DeepIndoorCrowd and have an direct insight into the temporal dynamic pattern of stores' crowds.

Figures 12a,b show the attention distribution for two catering stores. There is an obvious double peak pattern in the store's daily crowd flow. While forecasting the evening peak of a Saturday, the DeepIndoorCrowd mainly focuses on the inputs from the afternoon's peak of the current day. Besides that, the model also puts more attention

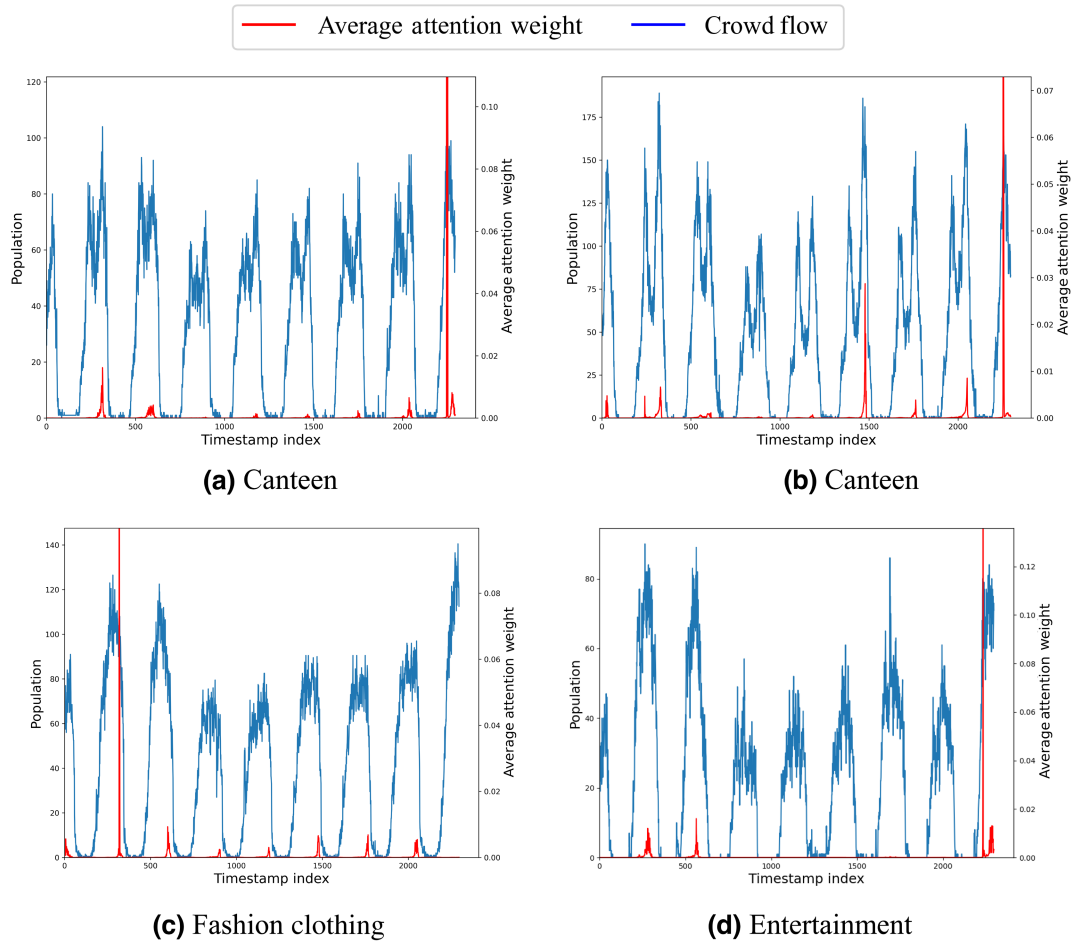

FIGURE 12 Attention weights distribution of four typical stores.

on the evening peak and afternoon peak of the previous weekend and the previous day. The distributions reveal that the peak hour's crowd flow is the most prominent pattern for a canteen and the two daily peaks are closely related to each other.

Figure 12c is the prediction of a famous fashion clothing store. The same day of the last week is assigned with the maximum attention. At the same time, the corresponding times of every previous day also contribute to the prediction. These phenomena illustrate that the weekly season is most important for a clothing store while the daily pattern also plays an important role.

Figure 12d is from the major entertainment store of the mall, which has the largest variation of crowd flow during the week. The attention is mainly assigned to the previous moments of the current day and the corresponding times of the weekend of last week. Entertainment places tend to have a larger gap between crowd flow on weekdays and weekends. The gap makes it difficult to make the prediction of weekends based on weekdays. So, the model chooses to put attention on the current day and weekly pattern to forecast the flow.

To understand how semantic features work in prediction, we visualized the embedding distribution of store's floors and serving types. The relationship between embedding locations can reveal the similarity of their crowd flow pattern. Analyzing the embedding distribution can help us better understand how semantic features works. The embedding distributions in the feature space are shown in Figure 13.

As shown in Figure 13a, the relationship between a store's serving type can be easily interpreted. Clothing stores are close to each other, which means their crowd patterns are similar. Moreover, the makeup store is closest to a jewelry store. This can be interpreted as their attracted crowd flows are from the same group of customers. At last, the canteen is distant to others, which can be attributed to its unique twin peak flow pattern. As for the store's floor, B1 and F3 are close to each other, which may be explained by both of them operate many canteens and the similarity between F1 and F4 is probably because the jewelry stores in F1 and female's clothing stores in F4 show similar crowd flow patterns.

5.8 | Computational complexity

We present the relative training time of each epoch and the number of converging epochs of FC-LSTM, LSTM-STF, LSTM-transformer and our model in Table 4. Considering the actual training time varies with different devices, we use the relative training time to evaluate the training consumption. It takes the minimum value of all as a measure and obtains the other values by calculating their multiple relationships. The converging epoch is defined as the epoch that evaluation metrics stop decreasing significantly or start rising on the test dataset.

Theoretically, the time complexity of the DeepIndoorCrowd is $O(t^2d + td^2 + d)$, where t is the length of the input time, and d is the dimension of the hidden units. We have to admit that the time complexity of our model in a single epoch is the highest among these models. Informer is specially designed with a ProbSparse self-attention mechanism, it achieves best performance in this time consumption comparison. As shown in the ablation study, the reason for models like LSTM-transformer, FC-LSTM and Informer can coverge in fewer iterations is that they are only designed to learn the temporal pattern in crowd flow series. Due to the

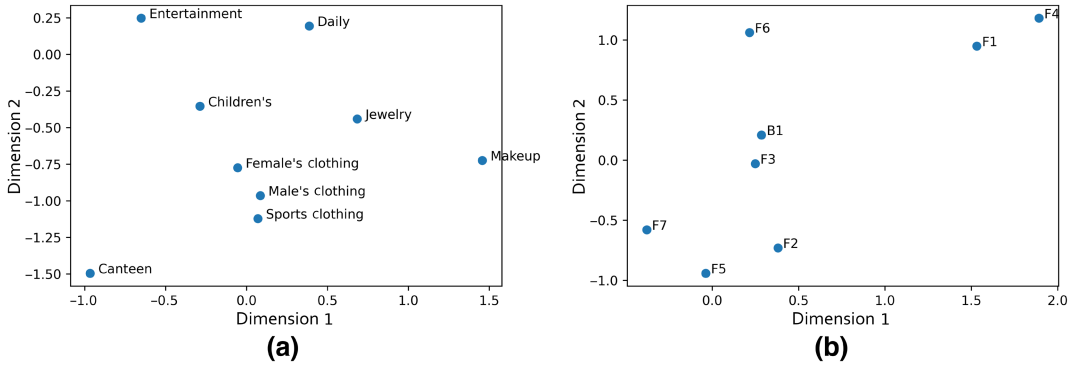

FIGURE 13 Embedding distributions of store's serving types (a) and floors (b).

TABLE 4 Training time complexity and converge epoch comparison between models.

Model	Relative training time per epoch (times)	Converge epoch (iteration)	Total training time (times)
Informer	1	10	10
FC-LSTM	1.23	10	12.38
LSTM-STF	1.28	19	24.40
LSTM-transformer	1.59	8	12.74
DeepIndoorCrowd	2.10	11	23.18

deficiency of semantic-temporal fusion, they start overfitting after converging. Compared with them, the DeepIndoorCrowd is more stable in gradient descent and can predict more accurately. Moreover, benefitting from the multi-term temporal dependencies capturing ability, the DeepIndoorCrowd can converge in fewer iterations compared with LSTM-STF which also has fused semantic information. The attention mechanism used in the DeepIndoorCrowd can directly capture the relationship between two distant timestamps, so the weekly flow pattern could be learned more quickly than others. Therefore, less training time is actually required in total.

6 | CONCLUSIONS

In this research, we proposed a novel indoor crowd flow prediction method named DeepIndoorCrowd. It models the indoor crowd flow from a semantic-temporal view. A semantic and temporal features fusion module named STF was proposed to integrate abundant indoor semantic features with sequential temporal features. With the feature fusion module, the model was able to involve the store's attribute and spatial features into prediction to learn the crowd flow pattern between stores with similar semantic characteristics. Then we further explored the indoor crowd flow data and proposed several typical temporal and semantic features. These features are believed to be correlated to indoor crowd flow, so they were expected to enable the model to predict the nonstationary dynamic of indoor crowd flow, which is important for an indoor shopping mall on holidays and weekends. Through conducting our experiment on a real-world dataset, the DeepIndoorCrowd was compared with other baseline methods, our model achieved a great improvement in all three evaluation metrics. Furthermore, we performed an ablation study to prove the effectiveness of semantic features and the multi-term temporal dependencies capturing ability of our model. And last, by analyzing the attention weights distribution produced in prediction, the model achieves interpretable predictions. We analyzed the weights' distribution of four typical stores and revealed the indoor crowd flow pattern for different shops. We found that the prediction of all kinds of stores relies on the weekly period pattern to different extents. The most prominent crowd flow pattern of a catering store in a day is its peak value, the generation of prediction is mostly based on these moments. While for the clothing store, the situation is different, the prediction is mainly based on its weekly pattern. For the entertainment store, caused by its usage, the crowd flow varies greatly in different days of the week. Generating its prediction heavily relies on the crowd flow of the previous few timestamps of the current day. We also visualized the embedding distribution of typical semantic features to explain why those semantic features work. Moreover, through comparing the relative time consumption in each training epoch and the number of converging epochs needed, we found out that even the complexity of our model is higher than others in a single epoch, the fusion of semantic features makes the model converge faster, so the total training time actually decreased. The modeling and featuring method we proposed is scalable to other indoor occasions. In future work, we are expecting to apply the model to more indoor environments.

ACKNOWLEDGMENTS

This work was supported by the National Key Research and Development Program of China, grant number 2021YFB3900803. We also thank the anonymous referees for their helpful comments and suggestions.

CONFLICT OF INTEREST STATEMENT

The authors declare that there is no conflict of interest that could have appeared to influence the work reported in this article.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Hengcai Zhang https://orcid.org/0000-0002-5004-9609

Peixiao Wang https://orcid.org/0000-0002-1209-6340

REFERENCES

Ahmed, M. S., & Cook, A. R. (1979). Analysis of freeway traffic time-series data by using box-Jenkins techniques. *Transportation Research Record*, *722*, 1–9. http://onlinepubs.trb.org/Onlinepubs/trr/1979/722/722-001.pdf

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. arXiv:1607.06450.

Cai, L., Janowicz, K., Mai, G., Yan, B., & Zhu, R. (2020). Traffic transformer: Capturing the continuity and periodicity of time series for traffic forecasting. *Transactions in GIS*, *24*(3), 736–755. https://doi.org/10.1111/tgis.12644

Cheng, S., Lu, F., Peng, P., & Wu, S. (2018). Short-term traffic forecasting: An adaptive ST-KNN model that considers spatial heterogeneity. *Computers, Environment and Urban Systems*, *71*, 186–198. https://doi.org/10.1016/j.compenvurbsys.2018.05.009

Cui, Z., Henrickson, K., Ke, R., & Wang, Y. (2020). Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. *IEEE Transactions on Intelligent Transportation Systems*, *21*(11), 4883–4894. https://doi.org/10.1109/TITS.2019.2950416

Dauphin, Y. N., Fan, A., Auli, M., & Grangier, D. (2017). Language modeling with gated convolutional networks. In *34th International Conference on Machine Learning*, *Sydney*, *NSW*, *Australia* (pp. 1–6). https://proceedings.mlr.press/v70/dauphin17a.html

Georgievska, S., Rutten, P., Amoraal, J., Ranguelova, E., Bakhshi, R., de Vries, B. L., Lees, M., & Klous, S. (2019). Detecting high indoor crowd density with Wi-fi localization: A statistical mechanics approach. *Journal of Big Data*, *6*(1), 1–23. https://doi.org/10.1186/s40537-019-0194-3

Guo, S., Lin, Y., Li, S., Chen, Z., & Wan, H. (2019). Deep spatial–temporal 3d convolutional neural networks for traffic data forecasting. *IEEE Transactions on Intelligent Transportation Systems*, *20*(10), 3913–3926. https://doi.org/10.1109/TITS.2019.2906365

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hosseini, K. S., Azaddel, M. H., Nourian, M. A., & Azirani, A. A. (2021). Improving multi-floor WiFi-based indoor positioning systems by fingerprint grouping. In *5th International Conference on Internet of Things and Applications (IoT)*, *Isfahan*, *Iran* (pp. 1–6).

Lee, J. (2004). A spatial Access-oriented implementation of a 3-D GIS topological data model for urban entities. *GeoInformatica*, *8*(3), 237–264. https://doi.org/10.1023/B:GEIN.0000034820.93914.d0

Li, H., Lu, H., Shou, L., Chen, G., & Chen, K. (2018). Finding most popular indoor semantic locations using uncertain mobility data. *IEEE Transactions on Knowledge and Data Engineering*, *31*(11), 2108–2123. https://doi.org/10.1109/TKDE.2018.2875096

Li, M., Lu, F., Zhang, H., & Chen, J. (2020). Predicting future locations of moving objects with deep fuzzy-LSTM networks. *Transportmetrica A: Transport Science*, *16*(1), 119–136. https://doi.org/10.1080/23249935.2018.1552334

Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2017). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. arXiv:1707.01926.

Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, *37*(4), 1748–1764. https://doi.org/10.1016/j.ijforecast.2021.03.012

Liu, F., Liu, J., Yin, Y., Wang, W., Hu, D., Chen, P., & Niu, Q. (2020). Survey on WiFi-based indoor positioning techniques. *IET Communications*, *14*(9), 1372–1383. https://doi.org/10.1049/iet-com.2019.1059

Liu, Z., Shi, W., Yu, Y., Chen, P., & Yu Chen, B. (2022). A LSTM-based approach for modelling the movement uncertainty of indoor trajectories with mobile sensing data. *International Journal of Applied Earth Observation and Geoinformation*, *108*, 102758. https://doi.org/10.1016/j.jag.2022.102758

Pennington, J., Socher, R., & Manning, C. D. (2014). GlovVe: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, *Doha*, *Qatar* (pp. 1532–1543). ACL.

Rasul, K., Seward, C., Schuster, I., & Vollgraf, R. (2021). Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *38th International Conference on Machine Learning*, *Proceedings of Machine Learning Research*, *Virtual Event* (pp. 1–12). Proceedings of Machine Learning Research (PMLR). https://proceedings.mlr.press/v139/rasul21a.html

Singh, U., Determe, J. F., Horlin, F., & Doncker, P. D. (2020). Crowd forecasting based on WiFi sensors and LSTM neural networks. *IEEE Transactions on Instrumentation and Measurement*, *69*(9), 6121–6131. https://doi.org/10.1109/TIM.2020.2969588

Sun, J., Zhang, J., Li, Q., Yi, X., Liang, Y., & Zheng, Y. (2022). Predicting citywide crowd flows in irregular regions using multi-view graph convolutional networks. *IEEE Transactions on Knowledge and Data Engineering*, *34*(5), 2348–2359. https://doi.org/10.1109/TKDE.2020.3008774

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *27th International Conference on Neural Information Processing*, Montreal, Canada (Vol. *2*, pp. 1–9).

Trivedi, A., Silverstein, K., Strubell, E., Shenoy, P., & Iyyer, M. (2021). WiFiMod: Transformer-based indoor human mobility modeling using passive sensing. In *4th ACM SIGCAS Conference on Computing and Sustainable Societies*, *Virtual Event*, *Australia* (pp. 126–137). ACM. https://doi.org/10.1145/3460112.3471951

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. arXiv:1706.03762.

Wang, P., Gao, F., Zhao, Y., Li, M., & Zhu, X. (2020). Detection of indoor high-density crowds via Wi-fi tracking data. *Sensors*, *20*(18), 5078. https://doi.org/10.3390/s20185078

Wang, P., Wang, H., Zhang, H., Lu, F., & Wu, S. (2019). A hybrid Markov and LSTM model for indoor location prediction. *IEEE Access*, *7*, 185928–185940. https://doi.org/10.1109/ACCESS.2019.2961559

Wang, P., Wu, S., Zhang, H., & Lu, F. (2019). Indoor location prediction method for shopping malls based on location sequence similarity. *ISPRS International Journal of Geo-Information*, *8*(11), 517. https://www.mdpi.com/2220-9964/8/11/517. https://doi.org/10.3390/ijgi8110517

Wang, P., Zhang, T., Zheng, Y., & Hu, T. (2022). A multi-view bidirectional spatiotemporal graph network for urban traffic flow imputation. *International Journal of Geographical Information Science*, *36*(6), 1231–1257. https://doi.org/10.1080/13658816.2022.2032081

Wang, P., Zhang, Y., Hu, T., & Zhang, T. (2023). Urban traffic flow prediction: A dynamic temporal graph network considering missing values. *International Journal of Geographical Information Science*, *37*(4), 885–912. https://doi.org/10.1080/13658816.2022.2146120

Wang, S., Cao, J., Chen, H., Peng, H., & Huang, Z. (2020). SeqST-GAN: Seq2Seq generative adversarial nets for multi-step urban crowd flow prediction. *ACM Transactions on Spatial Algorithms and Systems*, *6*(4), 22. https://doi.org/10.1145/3378889

Wang, Y., Yang, X., Zhao, Y., Liu, Y., & Cuthbert, L. (2013). Bluetooth positioning using RSSI and triangulation methods. In *10th Consumer Communications and Networking Conference (CCNC)* (pp. 837–842). IEEE.

Worboys, M. (2011). Modeling indoor space. In *3rd ACM SIGSPATIAL International Workshop on Indoor Spatial Awareness*, Chicago, IL (pp. 1–6). https://doi.org/10.1145/2077357.2077358

Zhang, J., Zheng, Y., & Qi, D. (2017). Deep spatio-temporal residual networks for citywide crowd flows prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, *31*(1), 1655–1661. https://doi.org/10.1609/aaai.v31i1.10735

Zhang, S., Deng, B., & Yang, D. (2023). CrowdTelescope: Wi-Fi-positioning-based multi-grained spatiotemporal crowd flow prediction for smart campus. *CCF Transactions on Pervasive Computing and Interaction*, *5*(1), 31–44. https://doi.org/10.1007/s42486-022-00121-6

Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., & Li, H. (2019). T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, *21*(9), 3848–3858. https://doi.org/10.1109/TITS.2019.2935152

Zheng, C., Fan, X., Wang, C., & Qi, J. (2020). GMAN: A graph multi-attention network for traffic prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(1), 1234–1241. https://doi.org/10.1609/aaai.v34i01.5477

Zheng, Y., Capra, L., Wolfson, O., & Yang, H. (2014). Urban computing: Concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology*, *5*(3), 38. https://doi.org/10.1145/2629592

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(12), 11106–11115. https://doi.org/10.1609/aaai.v35i12.17325

Zhu, J., Cheng, D., Zhang, W., Song, C., Chen, J., & Pei, T. (2021). A new approach to measuring the similarity of indoor semantic trajectories. *ISPRS International Journal of Geo-Information*, *10*(2), 90. https://doi.org/10.3390/ijgi10020090

How to cite this article: Chu, C., Zhang, H., Wang, P., & Lu, F. (2023). DeepIndoorCrowd: Predicting crowd flow in indoor shopping malls with an interpretable transformer network. *Transactions in GIS*, *00*, 1–25. https://doi.org/10.1111/tgis.13095