Prediction of Crowd Flow in City Complex with Missing Data

Shiyang Qiu 1 qsy1314@mail.ustc.edu.cn
Peng Xu 1 peng.xu1994@gmail.com
Wei Zheng 2 wei.zheng@comprehend.com.cn
Junjie Wang 1 sa516302@mail.ustc.edu.cn
Guo Yu 3 yuguo0405@qq.com
Mingyao Hou 2 houm@comprehend.com.cn
Hengchang Liu 1* hcliu@ustc.edu.cn
1. University of Science and Technology of China
2. Kehang Technology and Information
3. China People's Police University

Editors: Wee Sun Lee and Taiji Suzuki

Abstract

Crowd flow forecasting plays an important role in risk assessment and public safety. It is a difficult task due to complex spatial-temporal dependencies as well as missing values in data. A number of models are proposed to predict crowd flow on city-scale, yet the missing pattern in city complex environment is seldomly considered. We propose a crowd flow forecasting model, Imputed Spatial-Temporal Convolution network(ISTC) to accurately predict the crowd flow in large complex buildings. ISTC uses convolution layers, whose structures are configured by graphs, to model the spatial-temporal correlations. Meanwhile ISTC adds imputation layers to handle the missing data. We demonstrate our model on several real data sets collected from sensors in a large six-floor commercial complex building. The results show that ISTC outperforms the baseline methods and is capable of handling data with as much as 40% missing data.

Keywords: city complex, crowd flow forecasting, missing value, Graph Convolutional Network

1. Introduction

Crowd flow control is essential in public safety. When a large number of people rush in to a small region, events like stampede are more likely to happen.

November 11, 2007, 3 people were killed and more than 30 injured at the Supermarket Carrefour in Chongqing, China when the shop was offering 20% discounts on cooking oil [BBC News (2007)]. June 14, 2017, a fire broke out in the 24-storey Grenfell Tower and caused 72 deaths [Mynewsdesk (2017)]. December 31, 2014, a deadly stampede occurred in Shanghai, near Chen Yi Square on the Bund, where around 300,000 people had gathered for the new year celebration. In total, 36 people were killed and 49 injured in the stampede [BBC News (2014)]. September 17, 2018, five people are killed, and seven injured, when the crowd left CAF Champions League Quarterfinal Match in Luanda, Angola [Winning (2018)].

To prevent such accidents, public safety authorities usually implement strong emergency protocols in large building complexes, including police resource re-allocation and building exits controls, to prevent overly high crowd flow. In practice, the implementation of such protocols is usually costly as it requires a significant amount of additional police resource. In tradition, allocation of police resource is mainly based on empirical experience, which is subjective and error-prone. With the development of urban infrastructure, decisions can be made with knowledge of current crowd flow state. However, since it requires some time for the deployment to take place, the re-allocation strategy should be made with not only current condition, but also future state. Thus, it is extremely important to accurately predict high crowd flow ahead of time.

The prediction, however, is a difficult task because of complex spatial-temporal dependency in crowd flow and missing values. The future crowd flow is relevant to both short-term and long-term history in many nearby regions, which complicates the spatial-temporal dependencies. To capture the nonlinear time correlations, methods including RNN, LSTM, GRU are applied to crowd flow forecasting. As for spatial correlation, partitioning an large area into grids is a easy way to handle the spatial relation of regions, which is used in many studies on large metropolitan scales [Zhang et al. (2017); Zheng et al. (2018); Zhang et al. (2019)]. However, it can't be directly applied to the regions inside a building because of their 3d structures.

Modern large commercial building complex is typically consists of several skyscapers which may function as market, office building, hotel and apartment. There can be thousands of sensors collecting data from everywhere in the buildings. Malfunction of the sensors, manual system closure and network errors are inevitable in such complex systems. Most problems can be fixed automatically and only makes some random missing values. Some hardware failures, on the other hand, may require hours to be discovered and fixed. During this time period, the crowd flow data comes with a long period of missing values.

These missing values make the data difficult to explore with high efficiency. Most approaches use only valid data to train their model, which dramatically decreases the training set size and make the model less applicable in practice. Another solution is to infer the missing values according to its periodic characteristics, which is known as data imputation, with methods like temporal smoothing [Lipton et al. (2015)], or modified LSTM model [Tian et al. (2018)]. However, these methods use only temporal dependency, and are not able to capture the spatial correlations in crowd flow data.

The crowd flow data have obvious spatial correlations. For example, the sum of inflow and outflow for nearby regions are similar. We can utilize this feature to infer missing values even when a long period of data is missing.

In Impute Spatial-Temporal Convolution (ISTC) model, we use the missing patterns explicitly and impute missing values with spatial correlations. The impute layer is capable of handling the complex pattern of the spatial correlation and the graph nature of crowd flow data in city-complex. Then, convolution on both spatial and temporal dimensions are applied for future crowd prediction. The main contributions of this paper are as follows:

1. We propose a variant of GCN to impute missing crowd flow data by learning spatial correlations.

2. We use the proposed GCN variant to build a crowd flow forecasting model which can use data with missing values in both training and testing stages.

3. Experiments show that the proposed approach is able to achieve high accuracy with missing values in input data.

The remainder of the paper is as follows: Section 2 discusses related works on human mobility pattern, time-series forecasting, spatial-temporal data forecasting and prediction with missing values. Section 3 formulates the crowd flow prediction problem. Section 4 proposes the Impute Spatial-Temporal Convolution model. Section 5 shows our experimental results and section 6 concludes the paper.

2. Related Work

In this section, we will introduce studies involved in forecasting crowd flow and handling missing data.

2.1. Classical Models and Neural Networks for Time-series Forecasting

Forecasting crowd flow can be viewed as a time-series problem. Some approaches take advantage of statistical methods like auto-regressive integrated moving average model (ARIMA) [Moayedi and Masnadi-Shirazi (2008)], seasonal ARIMA [Williams and Hoel (2003)]. However, it's difficult for these approaches to effectively explore the nonlinear dependency in the time-series data. It's also a known challenging task to incorporate spatial-dependency in these time-series models.

Neural networks have been applied with great success in many fields such as computer vision [Redmon et al. (2016); Joo et al. (2018)], speech recognition [Graves et al. (2013); Amodei et al. (2016)], and natural language processing [Devlin et al. (2018)], etc. Models built on deep neural networks have also been used successfully for time-series data, including stacked autoencoder (SAE) [Wang (2015)], long short term memory(LSTM) [Lipton et al. (2015); Tian et al. (2018)] and gated recurrent unit (GRU) [Che et al. (2018)], etc. Although these models are capable of handling the complex patterns in time-series data, they lacks the ability to capture specific spatial relations of crowd flow.

2.2. Spatial-Temporal Data Forecasting

To model both the spatial and temporal dependencies of crowd flow, many approaches [Zhang et al. (2017); Zheng et al. (2018); Zhang et al. (2019)] split traffic network into rectangle grids and treat them as images, where each grid donates a pixel. However, these approaches ignore the irregular nature of geographical regions.

Piatkowski et al. (2013) and Hoang et al. (2016) use Markov Random Fields to model the spatial and temporal correlations. Similar to our work, Howard et al. (2017) tackle this problem using graph-CNNs. Regions are generated from segments segmented by roads, and crowd flow are calculated according to trajectories of individuals. However, neither of grid split nor road segment split can be directly applied to the regions inside a building because of their 3d structure.

Most of the previous works focus on spatial-temporal prediction in a large scale, e.g. city level. These works seldom consider the 3d structure of buildings, which is important for predicting in-building crowd flows. To the best of our knowledge, forecasting crowd flows has never been done at the scale of in-building level and in a data-driven way.

2.3. Prediction with missing values

Most approaches discard samples with missing values and use only valid data, which may cause significant loss of training data size. With the consideration of noisy and missing data, Hoang et al. (2016) use Intrinsic Gaussian Markov Random Fields (IGMRF) to model both the period flow and trend flow, which makes the model robust against noise and missing data. The missing pattern, however, is not take into account in the model design. Che et al. (2018) use informative missing patterns to make predictions based on Gated Recurrent Unit(GRU). They make use of missing patterns in the input features, but it is not suitable for time-series forecasting task where missing value appears both in input and output data. Tian et al. (2018) proposed a variant of LSTM which can be trained with missing values by explicitly combining the missing pattern. However, this method use only temporal dependency, and is not able to capture the spatial relations in crowd flow data.

3. Problem Statement

The goal in this research is to accurately predict future inflow and outflow in each region inside the shopping mall of a city complex.

Suzhou Center is a large scale city complex located in the heart of the city at Suzhou Industrial Park. It combines shopping malls, offices, residence buildings, movie theatres and hotels with a total construction area of 1.13 million sqm. Crowd flow data are gathered from numerous cameras located in the shopping mall of Suzhou Center.

The Shopping Mall has a gross floor area of 350,000 sqm, with over 1,000 shops, including flagship stores of international brands, fashion brands, shopping and leisure brands, children's entertainment, and cultural experiences. The combination of multi-purpose central courtyard, aerial corridor, building bridge, rooftop platform, basement passageway and observation terrace highlights the architecture but also raises significant concerns in public safety, especially in crowd flow monitoring, forecasting and control.

Fig. 1 shows sensors in first floor. Thousands of cameras have been installed in the shopping mall areas for surveillance purpose. Cameras near entrances and elevators are also used as crowd flow counter, with specific hardware and software installed.

We formulate the crowd flow forecasting problem as a spatial-temporal graph (STG) prediction problem. Our goal is to predict the future crowd inflows and outflows in each node of a STG based on historical observations and meta info. Such predictions can be sent to safety authorities in real time to provide crucial information regarding the safety situation in immediate future. Table 1 lists the notation used in the paper.

Definition 1 ***(STG)*** *A spatial-temporal graph (STG) denotes as $G = (V, A)$, where V is the N vertices in the graph, and $A \in \mathbb{R}^{N \times N}$ is a binary adjacency matrix. Each sensor is viewed as a vertex associated with time-varying inflow and outflow, and the edges represents*

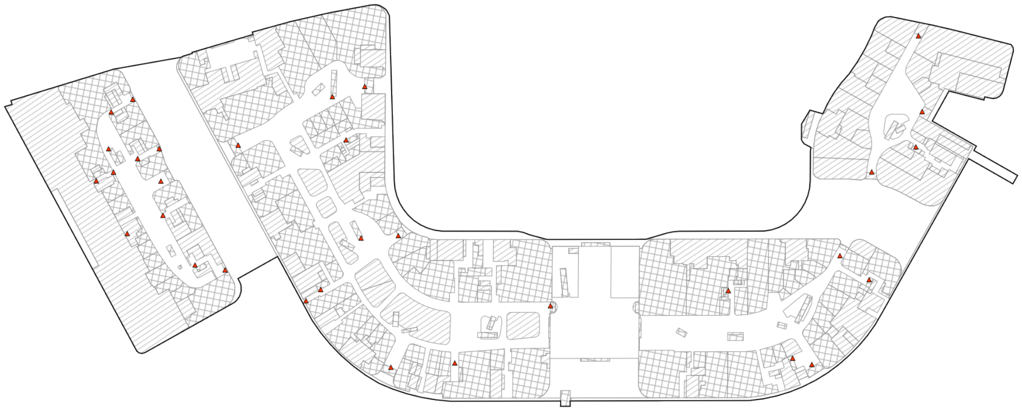

Figure 1: Sensors in first floor

Table 1: Notation

Symbol	Description
$G = (V, A)$	spatial-temporal graph
$V = \{v_i \mid i = 1, 2, 3, \cdots, N\}$	N sensors
$A \in \mathbb{R}^{N \times N}$	binary adjacency matrix
$p_i = (lat_i, lng_i)$	geographical position of sensor v_i
$floor_i$	floor index of sensor v_i
T	available timestamp set
$X_t \in \mathbb{R}^{N \times C}$	matrix of node feature vectors at timestamp $t \in T$

the connectivity of different sensors. Specifically, each vertex $v_i \in V$ has a geo-spatial position $p_i = (lat_i, lng_i)$ and a floor index $floor_i$, and time-varying attributes. These attributes at time t can be viewed as $X_t \in \mathbb{R}^{N \times C}$, where C is the count of attributes for each vertex.

As observation of each sensor have only direct connection with nearby sensors in the same floor and adjacent floors, we construct a static graph adjacency matrix A based on the geographical distance and floor index of each sensor as follows:

$$A_{ij} = \begin{cases} 1, \ \textit{if } i \neq j, |floor_i - floor_j| = 0, dist(p_i, p_j) < \kappa_0, \\ 1, \ \textit{if } i \neq j, |floor_i - floor_j| = 1, dist(p_i, p_j) < \kappa_1, \\ 0, \ \textit{otherwise}. \end{cases} \tag{1}$$

κ_0 and κ_1 are two parameters to determine the distance threshold of neighbors and control sparsity of the adjacency matrix.

Definition 2 ***(Missing mask)***

Every 10 minutes, each sensor in the shopping mall reports the accumulated inflow and outflow since midnight to the server. If the interval between a timestamp and latest record is larger than 15 minutes, data at this timestamp is marked as a missing value. The crowd flow is defined as the total number of people that pass through the specified entrance during a fixed period, i.e, the subtraction of accumulated crowd flow count at consecutive timestamps. The subtraction is marked as missing if any of the consecutive timestamps is missing.

Since the inflow and outflow are collected by one sensor, missing values always appears at same nodes in each timestamp. Thus, the missing mask at time stamp t can be represented as $M_t \in \{0,1\}^N$, where

$$M_t[i] = \begin{cases} 1, \textit{ if observation of node } v_i \textit{ at timestamp } t \textit{ is valid,} \\ 0, \textit{ if observation of node } v_i \textit{ at timestamp } t \textit{ is missing.} \end{cases} \quad (2)$$

Definition 3 ***(STG forecasting with missing values)*** *Given a graph $G = (V, A)$, history observation $\{X_t | t = 1, 2, \cdots, T\}$, and missing value mask $\{M_t | t = 1, 2, \cdots, T\}$, predict X_{T+1}.*

4. Method

4.1. Observation

Missing data in the crowd flow occurs for many different reasons, such as malfunction of the sensor, manual system closure and network errors. Regardless of the cause, we found that most missing values can be divided into two categories based on statistical analysis.

More than 70% of the consecutive missing values have missing duration less than 1 hour, but there are also many consecutive missing values that last for several hours, with the longest one duration of 23 hours and 20 minutes.

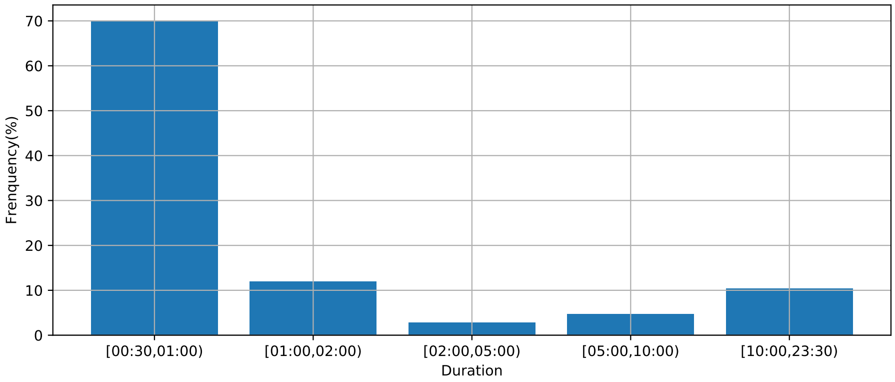

Figure 2: Duration of consecutive missing values

Fig 3 shows the distribution of cosine similarity between all node pairs. More than 60% of node pairs have a similarity higher than 0.7, which indicates a strong correlation between the nodes. Thus, we can impute the missing values based on the spatial correlation between different sensors. This imputation method can be applied to both random missing values and long-term missing values.

4.2. Impute Spatial Temporal Convolution

Based on these observations, we propose the Impute Spatial Temporal Convolution(ISTC) model to predict future crowd flow. The architecture of ISTC is introduced in Figure 4.

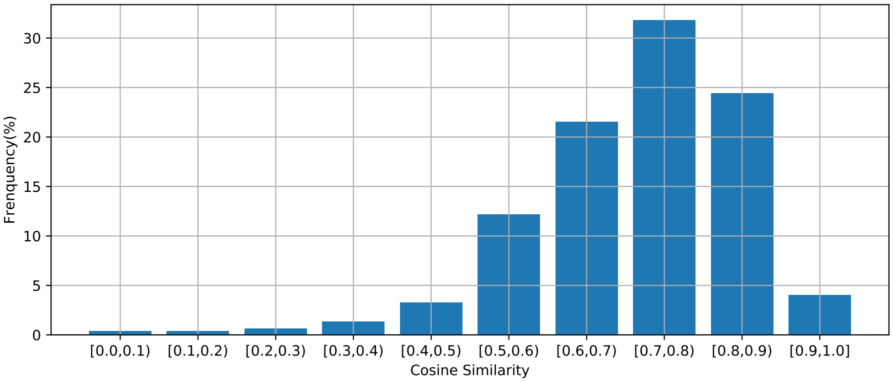

Figure 3: Distribution of Nodes Similarity

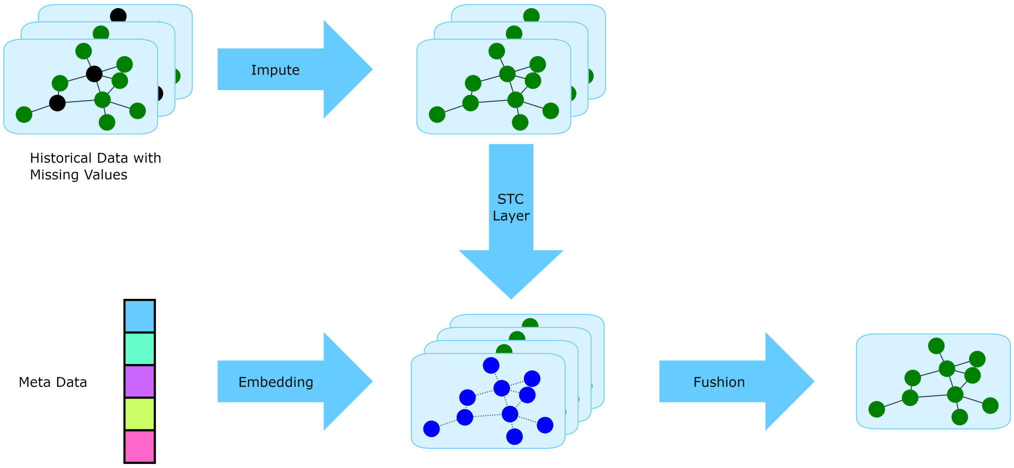

Figure 4: The structure of proposed Imputed Spatial-Temporal Convolution model.

This model imputes missing data with learned spatial correlation, and predicts future crowd flow with a temporal convolution layer. The model consists of three parts: imputation layer, Spatial-Temporal convolution, and fusion with embedded meta data.

4.3. Impute Layer

Assume that we are predicting crowd flow at timestamp T_F, the impute layer takes the historical data $Z^l = \{X_{T_F-1}, X_{T_F-2}, ..., X_{T_F-T_P}\} \in \mathbb{R}^{T_P \times N \times C}$ and corresponding missing mask $\{M_{T_F-1}, M_{T_F-2}, ..., M_{T_F-T_P}\}$ as inputs. T_P here means the length of input historical steps.

History data at each timestamp is concatenated with corresponding missing pattern, and the output of impute layer $X^{'}$ is calculated from history data and missing mask as:

$$X_t^{'} = \sigma\left(W_1 X_t^{\dagger} W_2\right) \tag{3}$$

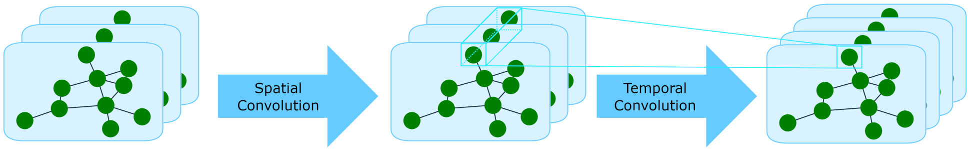

Figure 5: The structure of Spatial-Temporal Convolution, which consists of two convolutions applied on spatial and temporal dimensions respectively

where $X_t^{\dagger} = \text{concat}(X_t, M_t) \in \mathbb{R}^{T\times N\times(C+1)}$ is the concatenated input, $W_1 \in \mathbb{R}^{N\times N}$ and $W_2 \in \mathbb{R}^{(C+1)\times C}$ are trainable weights that donate the spatial correlation between different sensors and the affect of missing values respectively, and σ is the activate function, e.g. LeakyReLU.

The weight W is initialized as $W_0 = \text{concat}(I_C, 0_{1,C})$ so that the model does not use missing patterns at start, and learns from the missing pattern during training steps. I_C is the identity matrix, $0_{1,C}$ is zero matrix.

The same imputation is applied to each historical timestamp, results in an imputed historical data with the same shape as input historical data: $Z^{l+1} = \{X'_{T_F-1}, X'_{T_F-2}, ..., X'_{T_F-T_P}\} \in \mathbb{R}^{T_P\times N\times C}$.

4.4. Spatial-Temporal Convolution

In the Spatial-Temporal Convolution layer, we first fuse features of every nodes with graph convolutional network, then features of multiple timestamps are convolved in temporal dimension to extract the spatial-temporal feature representation of the graph.

4.4.1. Spatial Convolution

To handle spatial correlation, we use the graph convolutional network as in the work of Kipf and Welling (2016):

$$\star_G X = \sigma\left(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}XW^l\right) \tag{4}$$

where $\hat{A} = A + I_N$ is the adjacency matrix of G with added self-connections as in eq 1, $\hat{D} = \sum_j \hat{A}_{ij}$ is the degree matrix for $\hat{A}$, $W^l \in \mathbb{R}^{C_l\times C_{l+1}}$ is the trainable weight, and σ is the activate function, e.g. ReLU.

This layer captures 1-hop spatial correlations, and we stack multiple such layers to model multi-hop correlations and interactions. Just like the impute layer, graph convolutional network is also applied to each timestamp, results in an output $Z^l \in \mathbb{R}^{T_P\times N\times C_l}$

4.4.2. Temporal Convolution

After the spatial convolution, crowd flows are fused on the graph. The information across timestamps, however, is still isolated. To obtain spatial-temporal features, convolution with kernel K_l of size $[C_l, C_{l+1}, Q, 1]$ is performed on each node,

$$Z_i^{l+1} = Z_i^l * K_l \tag{5}$$

where $Z_i \in \mathbb{R}^{T_P \times C_l}$ represents extracted features of node i, and Q donates the size of time window. Then, a max-pooling is applied on the time dimension, make the STC layers capable of handling random missing values. Multiple such temporal convolutional layers are stacked to capture the complex temporal correlation.

4.5. Meta Data Embedding

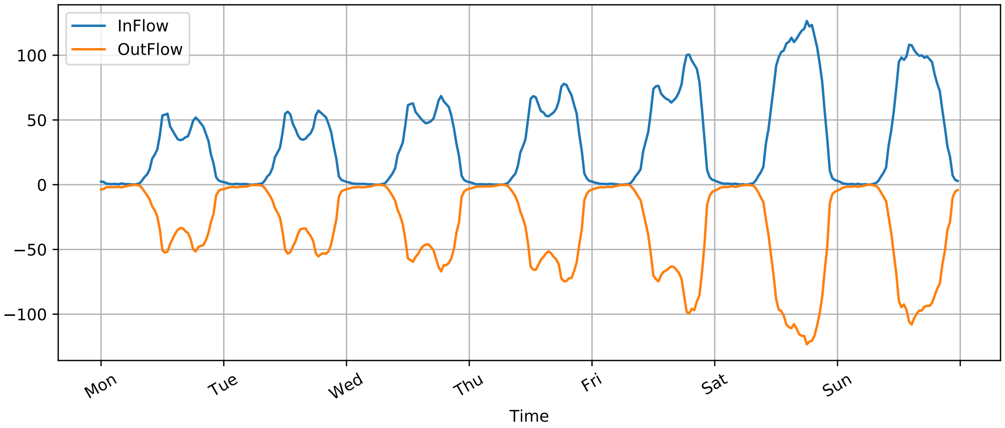

Figure 6: Weekly average crowd flow. X-axis is the index of 30 minutes interval time slots during a week. Y-axis is the average of all nodes at all historical time-slots.

As shown in Figure 6, the crowd flow pattern in weekdays and weekends are very different, and the time slot of a day also influence the crowd flow pattern greatly. To utilize this affect, we fuse the meta data (e.g. time of the day, day of the week) with a fusion layer.

The time of the day and day of the week are encoded into two class index, and then feed into two separate embedding layer whose embedding size are both the same as node number N, and then the outputs are stacked as $O_{meta} \in \mathbb{R}^{2 \times N}$, so that it can be concatenated with the output of STC layer. Then, O_{meta} is feeded into the fusion layer along with the output of STC layer.

Since the meta data is embedded to match the shape of prediction, we uses a full connected layer to produce the final forecasting crowd flow, with Relu activation to ensure non-negative predictions.

5. Evaluation

In this section, we compare the proposed approach with several baseline models in terms of effectiveness. The experimental setting, measurements, results and discussion of the experiments are provided.

5.1. Experimental Setting

Our crowd flow data set consists of data collected from February 15, 2019 to May 10, 2019. The data is resampled to time interval of 30 minutes. Since the mall of Suzhou Center is only open from 10:00 to 22:00, the data at night are meaningless zeros. Thus, we only keep the data from 9:00 to 23:00. After all these preprocessing steps, we get a dataset with 180 nodes and 2350 timestamps. These data comes with around 5% missing values.

We choose data from February 15, 2019 to April 30, 2019 as the train set, and data from May 01, 2019 to May 10, 2019 as test set. To evaluate the robustness of our model when more data are missing, two types of missing values are inserted into test set in evaluation, and the predictions are always compared to real data with 5% missing values.

Both the random missing dataset and long-term missing dataset are built upon aforementioned data, with extra missing values added in different ways. For random missing dataset, missing values are added randomly. For long-term missing dataset, missing values are added as follows: Choose a duration t_d from 5 hours to 20 hours randomly. Choose a sensor v_i and a start timestamp t_s from all possible choices randomly. Delete data of sensor v_i at timestamps $t \in T | t_s <= t <= t_s + t_d$.

Our model is build as mentioned in section 4, and it's implemented using PyTorch [Paszke et al. (2017)] and trained via backpropagation and Adam optimization [Kingma and Ba (2014)]. To train our model with missing values in input data, we only calculate the mse loss on valid data as:

$$\text{Masked MSELoss} = \frac{\sum\limits_{i=0}^{N} m_i \cdot (\hat{f}_i - f_i)^2}{\sum\limits_{i=0}^{N} m_i} \tag{6}$$

where N is the number of test sample, f_i is the real crowd flow, $\hat{f}_i$ donates the predicted value, and m_i is the binary mask of each value.

We use baselines including:

1. HA: Historical Average which use the weekly historical average at each time-slot as the future crowd flow. For example, the prediction for each 9:30 at Monday is the averaged crowd flows from all historical 9:30 at Mondays.

2. ARMA: ARMA model from python packages statsmodels. One model is trained for each feature on each nodes, with order selected by grid-search.

3. CNN: Convolutional Neural Network with no spatial convolution. We utilizes a two-layer CNN with the same structure as the temporal convolution layer in our ISTC model.

4. STC: Spatial-Temporal Convolution which is almost the same as ISTC model except for that the absence of impute layer.

We use the mean-absolute error (MAE), and the root-mean-squared error (RMSE) to evaluate the prediction accuracy. Both MAE and RMSE are also calculated with only valid data:

$$\begin{aligned} \mathrm{MAE} &= \frac{\sum_{i=0}^{N} m_i \cdot \left| \hat{f}_i - f_i \right|}{\sum_{i=0}^{N} m_i} \\ \mathrm{RMSE} &= \left(\frac{\sum_{i=0}^{N} m_i \cdot (\hat{f}_i - f_i)^2}{\sum_{i=0}^{N} m_i} \right)^{\frac{1}{2}} \end{aligned} \tag{7}$$

5.2. Results with added random missing values

Table 2: Evaluation results with added random missing values

Total missing	Metric	HA	ARMA	CNN	STC	ISTC
5%	RMSE	36.116	25.450	19.865	18.757	**16.663**
	MAE	16.188	13.653	10.932	10.127	**9.461**
10%	RMSE	36.296	25.877	20.724	19.373	**16.756**
	MAE	16.244	13.823	11.156	10.365	**9.536**
20%	RMSE	36.527	26.803	21.945	20.332	**17.534**
	MAE	16.347	14.196	11.649	10.832	**9.940**
40%	RMSE	37.387	30.402	26.086	22.230	**17.705**
	MAE	16.591	15.351	13.087	11.625	**10.048**
60%	RMSE	39.218	34.055	30.213	26.009	**20.716**
	MAE	17.217	16.995	15.462	13.372	**11.368**

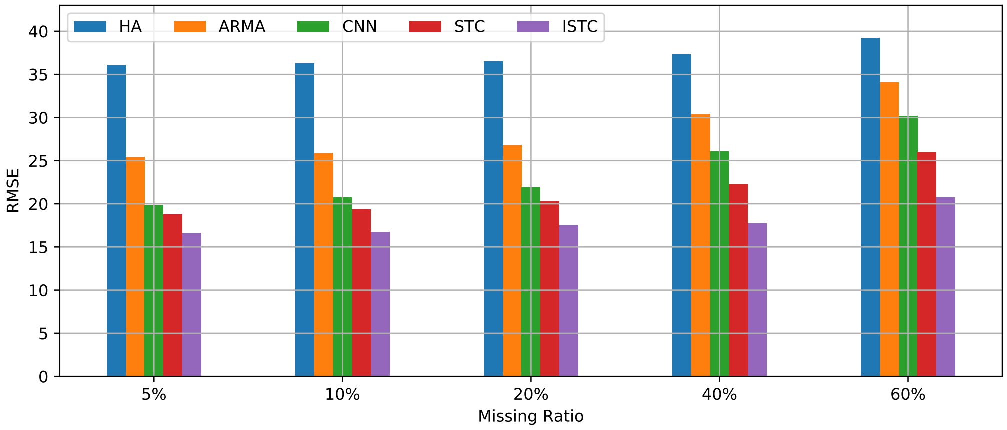

Figure 7: Evaluation results with added random missing values

Table 2 and figure 7 show the results when short-term missing values are added into the training and testing data. Our ISTC model has best performance at all missing rates.

The HA model has highest RMSE and MAE because it can not handle the temporal correlation in close history, nor the spatial correlation between different nodes. The ARMA model and CNN get better result compared to HA, with their ability to handle complex temporal correlations. And, with the ability to handle spatial correlation based on Graph Convolutional Network, both STC and ISTC outperforms other models that utilize only temporal relations. With the increase of missing ratio, there are fewer information which can be used in both training and predicting stage, and the forecasting accuracy therefore decreases, appears as increasing RMSE and MAE.

For ISTC model, the increase of RMSE with 40% missing values from 5% is 1.042, while the increase from 40% to 60% is 3.011, which is a remarkable difference. This difference shows the limitation of spatial correlation: when too many nodes are out of service, the ability of spatial convolution decreases. The RMSE of ISTC with 40% missing values(17.705) is even smaller than the RMSE of CNN model with 5% missing values, which shows the capability of spatial correlations when handling missing values in crowd flow data.

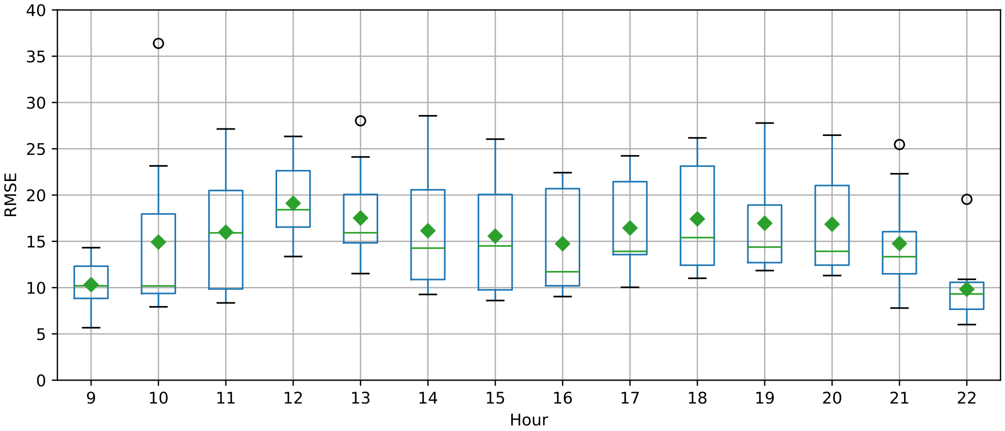

Figure 8: RMSE of ISTC during a day

Figure 8 shows the RMSE of our model during work hours of Suzhou-Center. Since there are few crowd flow at morning and night, the RMSE is relatively low at these periods. It also shows that RMSE near lunch and dinner time, e.g. around 12:00 and 18:00, is higher than afternoon, with Since cameras are installed at entrances, they are not able to capture crowd flow inside shops, which can be very high at lunch and dinner time when many people are inside the restaurants, the forecasting becomes more difficult.

5.3. Results with added long-term missing values

Table 3 and figure 9 show the evaluation results when long-term missing values are added into the training and testing data. And figure 10 gives a sample of the prediction results of CNN, STC and ISTC model with added long-term missing values. Our ISTC model still outperforms all other methods.

Unlike the result with random missing values, long-term missing values have much stronger influence on forecasting accuracy to all models except HA. Since the HA model

Table 3: Evaluation results with added long-term missing values

Total missing	Metric	HA	ARMA	CNN	STC	ISTC
5%	RMSE	36.116	25.450	19.865	18.757	**16.663**
	MAE	16.188	13.653	10.932	10.127	**9.461**
10%	RMSE	36.258	26.762	22.105	19.898	**17.638**
	MAE	16.238	14.077	11.629	10.592	**10.003**
20%	RMSE	36.020	31.401	27.954	25.854	**19.440**
	MAE	16.250	15.584	13.376	12.472	**11.060**
40%	RMSE	36.225	37.770	34.884	33.245	**22.001**
	MAE	16.426	18.047	16.235	15.585	**12.340**
60%	RMSE	38.796	45.985	44.502	42.323	**29.882**
	MAE	17.376	21.015	19.490	18.616	**15.672**

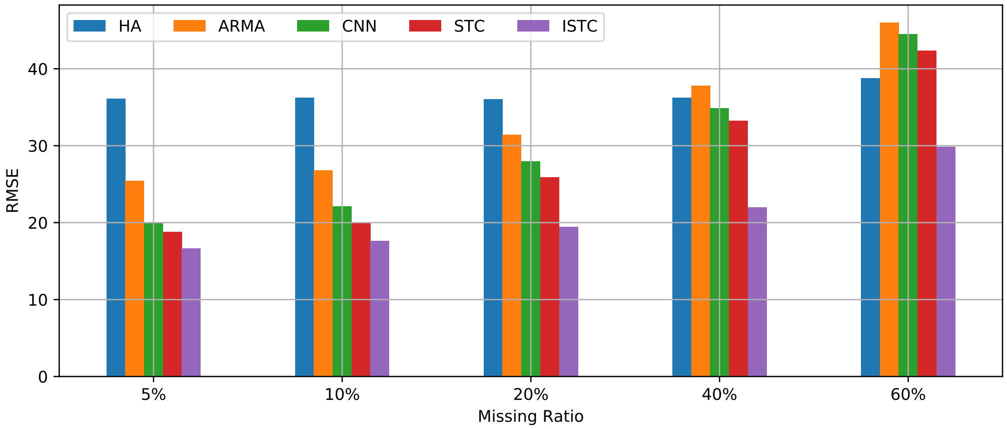

Figure 9: Evaluation results with added long-term missing values

use the whole historical data as input, which is much longer than longest missing duration, affect of these added missing values is similar to random missing values. All other models, including out ISTC model, however, accept historical data in a much shorter time-period as input, thus these long-term missing values have greater influence on these models. When the missing ratio reached 40%, the performance of ARMA becomes worse than HA. With 60% missing values, ARMA, CNN and STC get worse results than HA.

Figure 10 shows a sample prediction of CNN, STC and ISTC with long-term missing values. CNN model is greatly influenced by close history, thus long-term missing values make it gives a naive prediction, e.g. the horizontal line in figure 10, which indicates a failure of forecasting. The STC model manages to capture some trend from the learned spatial correlation, yet it is still greatly influenced by the missing in close historical data. It is very clear that ISTC model is the only one that predict the peak of crowd flow at the noon of May 13, when historical data since evening of May 12 are missing.

Our ISTC model, on the other hand, can minimize the influence of missing values with the knowledge of explicit missing patterns, and impute the missing data based on spatial

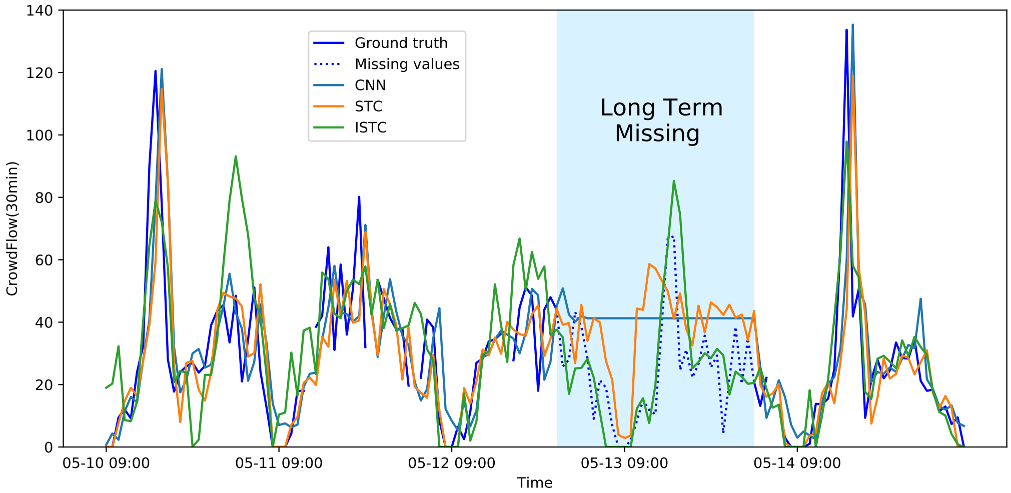

Figure 10: Prediction of CNN, STC and ISTC for inflow of node 0, with added long-term missing

correlation between nodes. With these ability, ISTC manages to keeps a relatively low RMSE with 40% missing values, and still outperforms all other methods for all missing ratios with added long-term missing values. The RMSE increase is 5.338 with 40% percent long-term missing values, while that of ARMA, CNN and STC are 12.320, 15.019, 14.488, respectively. Since long-term missing values are much rare than random missing-values, this is an acceptable result.

5.4. Discussion

Our ISTC model yields satisfactory results for both random and long-term missing values with as much as 40% total missing values. The RMSE improvement of ISTC is around 2.1 when original data with 5% missing values compared to STC model. With added random missing values, the RMSE improvement reaches 4.5 with 40% total missing. With added long-term missing values, the RMSE improvement reaches 11.2 with 40% total missing. This result shows that our ISTC model is capable of handling both random and long-term missing values.

The specifically designed impute layer explicitly uses the missing patterns and spatial correlations between vertices, making our model capable of handling long-term missing values.

6. Conclusion

In this paper, we propose an effective and efficient framework ISTC that can predict future in-building crowd flow with missing data. ISTC can use both temporal and spatial correlations, which makes it able to take advantage of the missing patterns explicitly. Experiments of our dataset shows our model outperforms all the baselines, and can give accurate prediction for both random and long-term missing values with as much as 40% total missing values.

References

Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In *International conference on machine learning*, pages 173–182, 2016.

BBC News. Three die in china sale stampede. http://news.bbc.co.uk/1/hi/world/asia-pacific/7088718.stm, 2007.

BBC News. Shanghai new year crush kills 35. https://www.bbc.co.uk/news/world-asia-china-30646918, 2014.

Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1):6085, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. IEEE, 2013.

Minh X Hoang, Yu Zheng, and Ambuj K Singh. Fccf: forecasting citywide crowd flows based on big data. In *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, page 6. ACM, 2016.

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8320–8329, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Zachary C Lipton, David C Kale, Charles Elkan, and Randall Wetzel. Learning to diagnose with lstm recurrent neural networks. *arXiv preprint arXiv:1511.03677*, 2015.

H Zare Moayedi and MA Masnadi-Shirazi. Arima model for network traffic prediction and anomaly detection. In *2008 International Symposium on Information Technology*, volume 4, pages 1–6. IEEE, 2008.

Mynewsdesk. Latest: Grenfell tower fire investigation, 2017. URL https://web.archive.org/web/20180620052605/http://news.met.police.uk/news/latest-grenfell-tower-fire-investigation-250453.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Nico Piatkowski, Sangkyun Lee, and Katharina Morik. Spatio-temporal random fields: compressible representation and distributed estimation. *Machine learning*, 93(1):115–139, 2013.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

Yan Tian, Kaili Zhang, Jianyuan Li, Xianxuan Lin, and Bailin Yang. Lstm-based traffic flow prediction with missing data. *Neurocomputing*, 318:297–305, 2018.

Zhanyi Wang. The applications of deep learning on traffic identification. *BlackHat USA*, 24, 2015.

Billy M Williams and Lester A Hoel. Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results. *Journal of transportation engineering*, 129(6):664–672, 2003.

Alexander Winning. Five die in stampede after angolan soccer match, 2018. URL https://www.reuters.com/article/us-angola-soccer/five-die-in-stampede-after-angolan-soccer-match-idUSKCN1LX0PV.

Junbo Zhang, Yu Zheng, and Dekang Qi. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Junbo Zhang, Yu Zheng, Junkai Sun, and Dekang Qi. Flow prediction in spatio-temporal networks based on multitask deep learning. *IEEE Transactions on Knowledge and Data Engineering*, 2019.

Zimu Zheng, Feng Wang, Dan Wang, and Liang Zhang. Buildings affect mobile patterns: developing a new urban mobility model. In *Proceedings of the 5th Conference on Systems for Built Environments*, pages 83–92. ACM, 2018.