Spatio-Temporal Graph Attention Embedding for Joint Crowd Flow and Transition Predictions: A Wi-Fi-based Mobility Case Study

XI YANG, University of Connecticut, USA
SUINING HE, University of Connecticut, USA
BING WANG, University of Connecticut, USA
MAHAN TABATABAIE, University of Connecticut, USA

Crowd mobility prediction, in particular, forecasting *flows* at and *transitions* across different locations, is essential for crowd analytics and management in spacious environments featured with large gathering. We propose GAEFT, a novel crowd mobility analytics system based on the multi-task graph attention neural network to forecast crowd flows (inflows/outflows) and transitions. Specifically, we leverage the collective and sanitized campus Wi-Fi association data provided by our university information technology service and conduct a relatable case study. Our comprehensive data analysis reveals the important challenges of sparsity and skewness, as well as the complex spatio-temporal variations within the crowd mobility data. Therefore, we design a novel spatio-temporal clustering method to group Wi-Fi access points (APs) with similar transition features, and obtain more regular mobility features for model inputs. We then propose an attention-based graph embedding design to capture the correlations among the crowd flows and transitions, and *jointly* predict the AP-level flows as well as transitions across buildings and clusters through a *multi-task* formulation. Extensive experimental studies using more than 28 million association records collected during 2020–2021 academic year validate the excellent accuracy of GAEFT in forecasting dynamic and complex crowd mobility.

CCS Concepts: • **Information systems** → *Mobile information processing systems.*

Additional Key Words and Phrases: Graph attention, Wi-Fi association data, crowd flow, transition, prediction.

ACM Reference Format:
Xi Yang, Suining He, Bing Wang, and Mahan Tabatabaie. 2021. Spatio-Temporal Graph Attention Embedding for Joint Crowd Flow and Transition Predictions: A Wi-Fi-based Mobility Case Study. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 4, Article 187 (December 2021), 24 pages. https://doi.org/10.1145/3495003

1 INTRODUCTION

Crowd mobility analytitcs have become increasingly important for spacious urban environments featured with large crowd gathering. The global crowd mobility analytics market is estimated to hit US$1,531 million by 2022[1]. Accurate and proactive crowd analytics can enable various ubiquitous computing applications, such as event surveillance [21], urban planning [46], epidemic and social analysis [28], recommendation and subsequent commercial promotions [45]. Especially, during the pandemic of COVID-19 [30], a crowd mobility analytics

[1] https://www.alliedmarketresearch.com/crowd-analytics-market

Authors' addresses: Xi Yang, xi.yang@uconn.edu, University of Connecticut, Department of Computer Science & Engineering, Storrs, CT, USA; Suining He, suining.he@uconn.edu, University of Connecticut, Department of Computer Science & Engineering, Storrs, CT, USA; Bing Wang, bing.wang@uconn.edu, University of Connecticut, Department of Computer Science & Engineering, Storrs, CT, USA; Mahan Tabatabaie, mahan.tabatabaie@uconn.edu, University of Connecticut, Department of Computer Science & Engineering, Storrs, CT, USA.

2474-9567/2021/12-ART187 $15.00
https://doi.org/10.1145/3495003

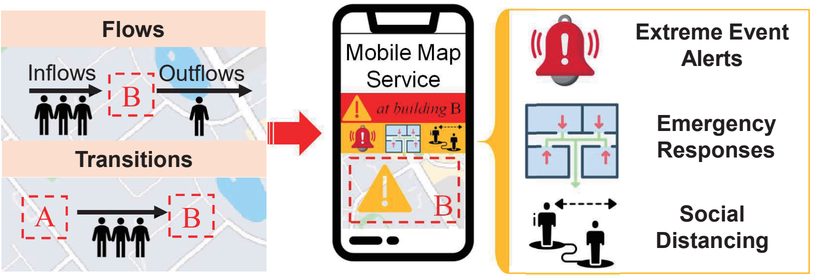


Fig. 1. Illustration of the crowd mobility analytics system applications.

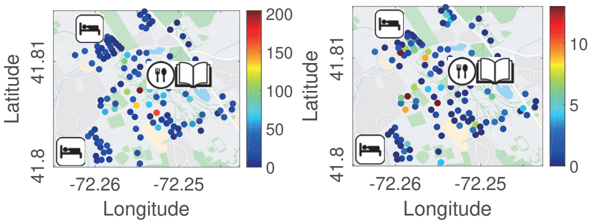
system can monitor and control the crowd distributions at many spacious sites, and subsequently help mitigate the spread of epidemic diseases.

In this paper, we develop a *predictive* crowd mobility analytics system to forecast the crowd mobility at different locations of a target site. As illustrated in Fig. 1, we focus on two important crowd mobility patterns, crowd *transitions* (number of people travelling from one location to another) and crowd *flows* (number of people entering or leaving a location, referred to as inflows/outflows). This predictive system can assist the relevant stakeholders (*e.g.,* crowd management and emergency departments) in handling potentially congested areas [1, 2] and enhancing their preparedness to potentially abnormal crowd distributions [4, 47], such as providing event alerts, deploying emergency responses, and enforcing social distancing. Using mobile apps or web-based services, these alerts as well as emergency and social distancing information can reach the public quickly.

As a *case study*, the predictive crowd mobility analytics system is developed for our university campus using the Wi-Fi association data collected from the campus network. We note that Wi-Fi access points (APs) have been ubiquitously deployed on our campus (like many other universities) to provide excellent Internet connection services for students, faculty, and staff. When users are connected to the Wi-Fi network, their locations can be approximated by the locations of the APs that their mobile devices are associated with (since the devices are associated with nearby APs for Internet access). Therefore, Wi-Fi association data can be used to identify the approximate locations of the end users in real time. This approach of sensing the crowds has the following two advantages. *First*, harvesting the Wi-Fi associations is highly automatic and non-intrusive — it leverages campus network infrastructures, requiring no apps downloaded on the end-user devices. *Second*, since no sensitive information (such as user IDs, MAC addresses, or IP addresses) is included in the analysis, our approach of using passive Wi-Fi association data (routinely collected by the university network services) is much less privacy intrusive compared to user ID card access records, camera-based [11], and other active Wi-Fi probe-based approaches [33]. Through partnership with our University Information Technology Services (UITS), our developed crowd mobility analytics system will assist the university campus management department in timely response to the future campus reopening and crowd gathering.

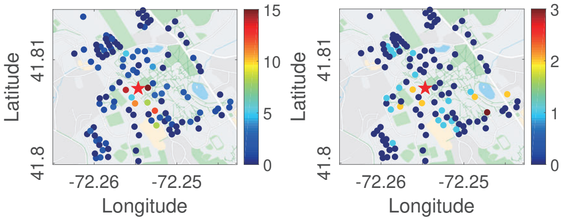
Conducting comprehensive crowd mobility analytics of the Wi-Fi data (Sec. 2.2), we have identified data *sparsity* and *skewness* as a major issue in the harvested crowd mobility data, *i.e.,* the majority of the crowd mobility data (in/out flows and transitions) are recorded at a few locations and in a few time periods. This poses two major challenges to practical deployment of crowd mobility analytics systems.

(1) *Spatio-Temporal Complexity Challenge*: Due to end users' complex daily routines and preferences, the crowd mobility sparsity and skewness vary spatially and temporally across different campus locations and time periods. Taking our campus as an example, we show in Fig. 2 the distributions of the inflows, *i.e.,* the number of people arriving at these labeled locations (campus buildings). We can see that the majority of the daytime activities (Fig. 2a) concentrate in the center of the campus (due to major academic and dining



(a) 11 am - 12 pm (b) 0 am - 1 am

Fig. 2. Total number of inflows at each building on Monday 2021-03-01 (warmer colors represent larger inflows).



(a) 11 am - 12 pm (b) 22 pm - 23 pm

Fig. 3. Variations of transitions from the campus center (red star) to other locations on Monday 2021-03-01 (warmer colors represent larger transitions).

activities), while in the evening the majority of crowd inflows are observed in the peripheral areas of the campus where most of the student residential halls are situated (Fig. 2b). Similar sparsity can be observed from Fig. 3, from which we can see the highly skewed transitions from the campus center to other campus locations in the daytime (mainly towards the center of campus in Fig. 3a) and at night (mainly towards the peripherals of the campus in Fig. 3b). Such sparsity variations across space and time make accurate modeling and prediction difficult.

(2) *Model Learnability Challenge*: An even more pressing issue for the crowd mobility analytics is the model learnability. Conventional deep learning prediction of complex crowd mobility [24, 43, 48] often requires dense features available within the crowd mobility data, which, however, might not be always valid given the above sparse and skewed data. Furthermore, the *fine-grained* crowd mobility analytics often require space and time discretization of high granularity (*e.g.,* predicting *hourly* crowd flows under different *APs*). It often causes the input features to be even more sparse and skewed, making it highly challenging for the conventional deep learning models [24, 27, 34, 42, 43, 48] to effectively learn and predict.

To address these challenges, we propose **GAEFT**, a novel crowd analytics system based on Graph Attention Embedding neural network and joint crowd Flow-Transition learning. GAEFT aims at *jointly* predicting two aspects of the crowd mobility: *(i)* the crowd *transitions*, *i.e.,* the transitions across the campus buildings and regions, and *(ii)* crowd *flows*, *i.e.,* the numbers of *arrivals* and *departures*, based on the association and disassociation records from the Wi-Fi APs. Our study makes the following three major contributions:

(1) *Comprehensive Crowd Flow and Transition Data Analytics*. We have conducted comprehensive studies of the crowd flows and transitions on campus, which motivate the adoption of a novel spatio-temporal clustering method to handle data sparsity. Specifically, we have designed a novel affinity propagation clustering method based on the spatial closeness and temporal transition connectivity across buildings, and group the APs in different buildings into clusters. The resulting more regular mobility patterns serve as one of the model inputs, and help mitigate the data sparsity and augment the model learnability of GAEFT.
(2) *A Novel Multi-task Learning Framework with Graph Attention Embedding*. To further handle the sparsity within the crowd mobility data, we propose a novel graph attention embedding design which incorporates the spatio-temporal correlations of the building neighborhoods (clusters) to enhance the GAEFT's learnability on crowd flow and transition. To further handle the data sparsity issue of the transition matrix, we generate two separate embeddings for inflows and outflows, and leverage two separate graph embedding learning modules to integrate them for the final transition predictions. Within each graph embedding learning module, we design a novel spatio-temporal multi-head attention mechanism that captures and differentiates the importance of the spatio-temporal correlations across the crowd flows and transitions. We then *jointly*

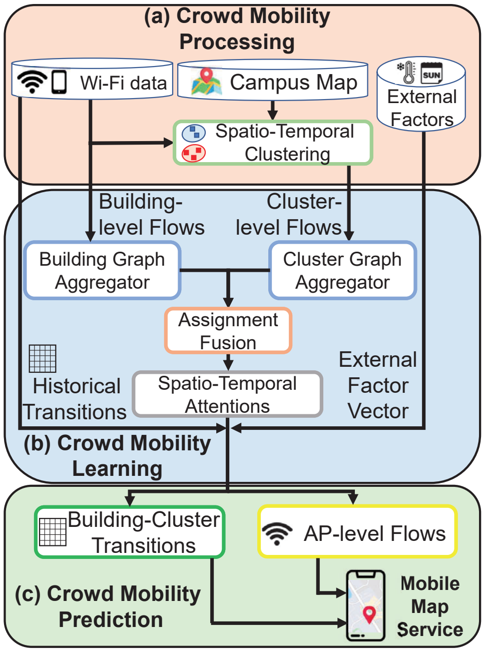


Fig. 4. Information flow of GAEFT: (a) crowd mobility processing; (b) crowd mobility learning; and (c) crowd mobility prediction.

predict the transitions and flows in a multi-task learning paradigm, which has been shown to enhance the prediction accuracy.

(3) *Extensive Real-world Experimental Studies.* Collaborating with our university information technology services (UITS), we conduct extensive experimental studies on the real-world crowd transitions and flows generated during 2020–2021 academic year (AY). Results from the extensive experimental evaluations demonstrate that our proposed model achieves higher accuracy in predicting the crowd flows (AP-level) and transitions (across buildings and clusters) than the state-of-the-art baseline models [24, 27, 34, 42, 43, 48].

System Overview: We illustrate the information flow of GAEFT in Fig. 4, which consists of three phases.

(**a**) In the *Crowd Mobility Processing* phase, we collect the Wi-Fi association data and find the crowd flows (in/outflows based on the associations and disassociations) at different locations covered by the Wi-Fi APs. Using the campus map, we group the buildings and their APs into clusters based on their *closeness* in terms of geographic locations as well as *connectivity* in terms of crowd transitions. We also collect and process the external factors (*e.g.,* weather conditions and weekdays/weekends) to assist the mobility prediction. Summarizing the above, we form the *building-level flows*, *cluster-level flows*, and *external factor vector* as inputs to GAEFT.

(**b**) In the *Crowd Mobility Learning* phase, GAEFT considers the target site such as the campus as a *network* with buildings or clusters as *nodes* and crowd transitions between nodes as *edges*. Based on the network graph thus formed, GAEFT takes in building/cluster-level flows and extracts the spatial features of the buildings/clusters using the building and cluster *graph aggregators*. The extracted cluster-level inflow/outflow embeddings are fused with the extracted building-level inflow/outflow embeddings using *assignment fusion*. We further enhance the extraction of spatio-temporal charateristics of the fused building-level inflow/outflow embeddings by *spatio-temporal attentions*. We then combine the building-level inflow/outflow embeddings with external factors as well as the historical transitions (in the same historical time periods).

The outputs will be simultaneously mapped towards transitions across the buildings and clusters, as well as the AP-level flows, through a multi-task learning mechanism. This way, we train GAEFT to learn and predict the sparse crowd mobility data.

(**c**) In the *Crowd Mobility Prediction* phase, given the trained GAEFT model, we predict the transitions and flows for the crowd management departments and relevant stakeholders. For instance, we can visualize the potential congestions and crowded spots on the mobile map service, and help inform the management departments and crowds for potential responses and preparations. Our fine-grained AP-level prediction will also enable localized crowd control [39] at different campus locations.

Societal Implications: Our campus crowd mobility studies are timely and important for providing guidance on predictive crowd flow and transition modeling and management. Since the outbreak of COVID-19, more than 700,000 cases[2] have been reported on college campuses in the U.S. by July 2021. Especially given the growing threats from SARS-COV2 variants [30], many university campuses in North America and Europe are facing unprecedented challenges in the coming 2021 Fall reopening. Although we use Wi-Fi association data for our prototype studies here, the insights and models from our study can be extended to other spacious urban environments (such as shopping malls), and other existing or emerging crowd sensing modalities, such as leveraging cellular signals [35] and camera tracking [11].

We organize the rest of the paper as follows. We first overview the datasets used, define important concepts, and present our data analysis and motivations in Sec. 2. Then, we present the detailed core formulation of GAEFT in Sec. 3 and the model integration and multi-task learning in Sec. 4. Afterwards, we present the experimental evaluation of GAEFT in Sec. 5. We then review the related work in Sec. 6, discuss the deployment of GAEFT in Sec. 7, and finally conclude in Sec. 8.

2 SYSTEM OVERVIEW, DATASETS & IMPORTANT CONCEPTS

We first present the Wi-Fi association dataset and other external factors considered in GAEFT in Sec. 2.1, followed by the definitions of important concepts and motivations in Sec. 2.2.

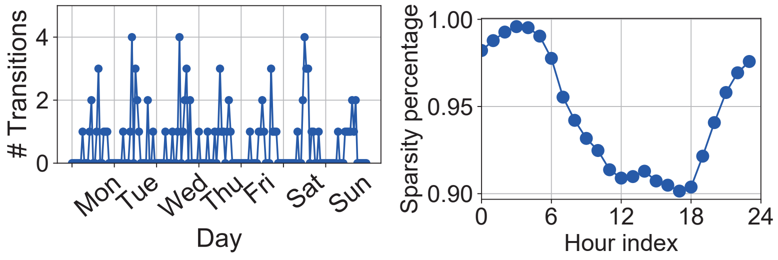
2.1 Dataset Overview

We have collaborated with our university information technology services to collect *Wi-Fi association data* from the campus network. To summarize, we have collected 28,477,044 Wi-Fi association records in total from 1,257 APs during 2020-10-11 to 2020-11-10 (Fall) and 2021-02-02 to 2021-04-10 (Spring). Specifically, we use a server that periodically (on a hourly basis) retrieves AP association and dissociation events from all the campus APs using standard network protocols [9]. Each Wi-Fi association record contains the following key attributes: user IDs (encrypted and sanitized), association timestamp and duration, and the MAC address of the AP for the association. Table 1 shows an example of a Wi-Fi association record. The user IDs have been encrypted and randomized before our data analysis for privacy protection. A user might have multiple mobile devices, such as a smartphone and a laptop, and these devices might associate with the APs under the same ID. By mapping multiple devices towards the single user, we can differentiate the users in the crowds. The IDs are discarded immediately after the mapping and aggregation. Based on the mapped IDs, we have identified 22,298 users in total who have association activities within our collected data. By examining the association records of two consecutive APs at two buildings where the users have visited, we infer the collective (aggregate) transitions across different campus buildings.

Table 1. An example of a Wi-Fi asssociation record.

User IDs	Association Timestamp (yyyy-MM-dd HH:mm:ss)	Session Duration	MAC of AP
Anonymized	2020-11-10 13:00:00	3 min 5 sec	12-digit hexidecimal number

[2] https://www.nytimes.com/interactive/2021/us/college-covid-tracker.html



(a) Dining hall → student center.

(b) Percentage of sparsity.

Fig. 5. (a) Time series of transitions from the dining hall to the student center; (b) percentage of sparsity in hourly total transitions.

In addition, we have considered other external factors such as weather conditions and weekday time which influence the crowd mobility. Specifically, we collect the temperature and hourly precipitation from the open data sources[3]. We collect 7,655 records of weather conditions for the period of 2020-10 to 2021-05 (including temperature and precipitation). Then we concatenate the hourly temperature (from 0 to 85° F), hourly precipitation (from 0 to 0.29 inches), and indicator of whether it is a weekday (denoted as 1) or not (denoted as 0), and form the external factor vector input, denoted as $\mathbf{e}$, for GAEFT. For instance, we have $\mathbf{e} = [10°\text{F}, 0.29 \text{ inch}, 1]$ for a certain time interval. We then normalize each dimension in $\mathbf{e}$ using min-max normalization.

2.2 Important Concepts and Motivations

We define the following important concepts and motivate our data-driven model designs.

Time Discretization. With the collective Wi-Fi association and disassociation records, we are able to capture the mobility (transitions and flows) at different campus locations with Wi-Fi AP coverage. For ease of modeling transitions/flows, we first discretize the time domain into equal-duration slots or intervals (1 hour in our study), each of which is indexed by k. In our data analytics, we observe that users' transition time may cover more than one time intervals. For instance, a user disassociated from an AP in a building at, say, 11:55am, then traveled for 15min and connected to an AP in another building at 12:10pm. Therefore, we investigate the transitions as well as flows for every two consecutive intervals as described below.

Crowd Flows. Given the Wi-Fi associations and disassociations at the campus Wi-Fi APs, we first define the crowd flows for GAEFT's formulation. Specifically, based on the encrypted user IDs, we map the Wi-Fi associations and disassociations at all $N^{(\mathrm{A})}$ Wi-Fi APs ("A" for APs) towards the numbers of user arrivals (inflows) and departures (outflows) at the time interval k, *i.e.,* the AP-level crowd flows $\mathbf{F}_k^{(\mathrm{A})} \in \mathbb{R}^{N^{(\mathrm{A})}\times 2}$. Note that we let the elements at $\mathbf{F}_k^{(\mathrm{A})}[:, 1]$ be the AP-level inflows and the ones at $\mathbf{F}_k^{(\mathrm{A})}[:, 2]$ be the AP-level outflows. Based on the AP-level crowd flow, we can aggregate the APs within the same building on the campus, and obtain the building-level crowd flows for all $N^{(B)}$ buildings ("B" for buildings), denoted as $\mathbf{F}_k^{(\mathrm{B})} \in \mathbb{R}^{N^{(\mathrm{B})}\times 2}$.

Crowd Transitions. Based on the building-level crowd flows, we then find the building-to-building transition matrix $\mathbf{T}_k \in \mathbb{R}^{N^{(\mathrm{B})}\times N^{(\mathrm{B})}}$, where each element $\mathbf{T}_k[b, b']$ $(b, b' \in \{1, \dots, N^{(\mathrm{B})}\})$ represents the aggregate number of clients who depart from buildings b in the time interval k to b' in either the interval k or $(k + 1)$ (without connecting to an AP in a third building in between). Based on our transition time analysis during 2020 Fall and 2021 Spring, we note that over 85% of the transitions last shorter than 2h. Therefore, we consider 1h for each time interval, and take into account the transitions which span at most two consecutive time intervals (k and $k + 1$).

[3] api.weather.com

Data Sparsity and Skewness. Through our data analysis of the Wi-Fi-based crowd mobility data, we have identified and quantified the sparsity and skewness as follows. We first show an example of the transitions from a dining hall to the student center within a week in Fig. 5a. The highly sparse, irregular, and dynamic transitions make it difficult to model and predict. We further show in Fig. 5b the temporal sparsity in the hourly transition matrices across buildings. We find the average percentage of sparsity in the transition matrices of all days. While the sparsity drops during the daytime due to more crowd mobility recorded, we can still observe significant zeros (over 90%) from the transition matrices, which makes it highly challenging to model and predict.

Spatio-Temporal Clustering to Mitigate Sparsity. Motivated by the above observations, we group the buildings on campus into different clusters based on the buildings' spatio-temporal mobility patterns. The buildings in a cluster form a region with similar mobility features. The periodicity and regularity of the transition patterns between buildings and clusters becomes clearer and thus potentially more learnable.

Our goal is to cluster the buildings into regions with similar spatio-temporal mobility patterns. To this end, we design a similarity score to accommodate two important perspectives: *(i)* the spatial similarity in terms of *geographic distances*, and *(ii)* the temporal similarity in terms of *historical transitions*. For *(i)*, we consider geo-distance (km) between two buildings b and b', denoted as $g[b, b']$. For *(ii)*, we conduct the min-max normalization upon the transition matrix $\mathbf{T}_k$ to obtain $\overline{\mathbf{T}}_k$ which is of similar magnitude as $g[b, b']$. We then define $\mathrm{diff}[b, b']$ as the squared average difference of all normalized transitions starting from and ending at buildings b and b', *i.e.*,

$$\mathrm{diff}[b, b'] = \frac{1}{N^{(\mathrm{B})}} \sum_{k} \sum_{b''}^{N^{(\mathrm{B})}} \left(\overline{\mathbf{T}}_k[b, b''] - \overline{\mathbf{T}}_k[b', b'']\right)^2 + \frac{1}{N^{(\mathrm{B})}} \sum_{k} \sum_{b''}^{N^{(\mathrm{B})}} \left(\overline{\mathbf{T}}_k[b'', b] - \overline{\mathbf{T}}_k[b'', b']\right)^2. \tag{1}$$

Our similarity score between two buildings, b and b', then becomes

$$\mathrm{sim}[b, b'] = -\beta \cdot g[b, b'] - \mathrm{diff}[b, b'], \tag{2}$$

where β is a hyperparameter. We then adopt the Affinity Propagation clustering [6] to cluster the buildings, which requires no explicit input of the number of clusters. We illustrate in Fig. 6 the resulting 21 clusters from 128 selected buildings based on all the transitions during 2020–2021 AY. We further show the resulting hourly transitions in Fig. 7 from the dining hall to one cluster, and we can observe more regular patterns than in Fig. 5a.

Cluster-level Flows and Building-Cluster Transitions. Given the $N^{(\mathrm{C})}$ clusters generated ("C" for clusters), we further aggregate and find the cluster-level flows at all clusters at the time interval k, $\mathbf{F}_k^{(\mathrm{C})} \in \mathbb{R}^{N^{(\mathrm{C})} \times 2}$, as the numbers of association records (inflow) or dissociation records (outflow) in all buildings of each cluster. Let c be the index of a cluster. Based on the formed clusters, we find the building-cluster transition matrix $\mathcal{T}_k \in \mathbb{R}^{N^{(\mathrm{B})} \times N^{(\mathrm{C})} \times 2}$ at the time interval k, where $\mathcal{T}_k[b, c, 1]$ represents the number of transitions from a building b to a cluster c, and $\mathcal{T}_k[b, c, 2]$ represents the number of transitions from a cluster c to a building b.

Based on the generated building clusters, we can obtain more regular crowd flow/transition patterns, mitigate the adversarial effects of the sparse and skewed crowd mobility data, and enhance the learnability of GAEFT. Furthermore, the knowledge of transitions between buildings and clusters consisting of multiple other buildings will help the relevant stakeholders to handle potentially congested areas for the spacious site management.

Problem Statement. Based on the processed data, we present the research problem for GAEFT as follows. Given the historical K *building-level flows* and *cluster-level flows*, $\{\mathbf{F}_k^{(\mathrm{B})}\}$ and $\{\mathbf{F}_k^{(\mathrm{C})}\}$, $k \in \{1, 2, \ldots, K\}$, the campus map and the graph network formed by the buildings, and the external factors $\mathbf{e}$, GAEFT simultaneously predicts the building-cluster transitions, $\widehat{\mathcal{T}}$, and the AP-level flows, $\widehat{\mathbf{F}}^{(\mathrm{A})}$, at the next target time interval.

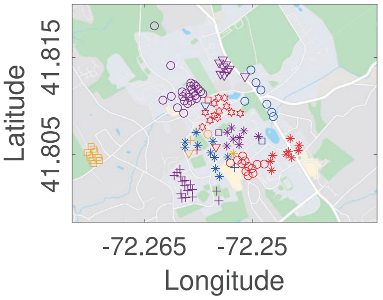


Fig. 6. Illustration of building clusters.

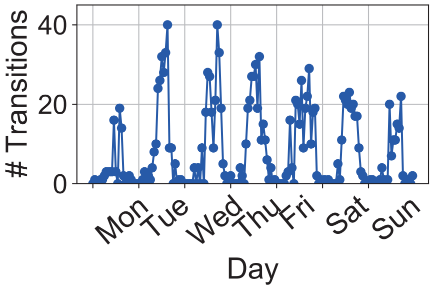


Fig. 7. Time series of hourly transitions from the dining hall to one cluster.

3 CORE FORMULATION OF GAEFT

We first overview the core model in Sec. 3.1, and present the designs of graph aggregator in Sec. 3.2. After that, we present how to fuse the cluster-level and building-level inflow/outflow embeddings in Sec. 3.3, followed by the spatio-temporal attention mechanisms in Sec. 3.4.

3.1 Overview of GAEFT Architecture

Fig. 8 overviews the core architecture of GAEFT. GAEFT consists of two parallel and connected pipelines in *jointly* learning the inflows (upper half in Fig. 8) and outflows (lower half in Fig. 8). The key idea of GAEFT is similar to the matrix decomposition where a sparse matrix is factorized into two dense one which preserve the most important information from the sparse inputs [19], and hence enhance the model learnability upon the sparse mobility data. We will present the following four major components.

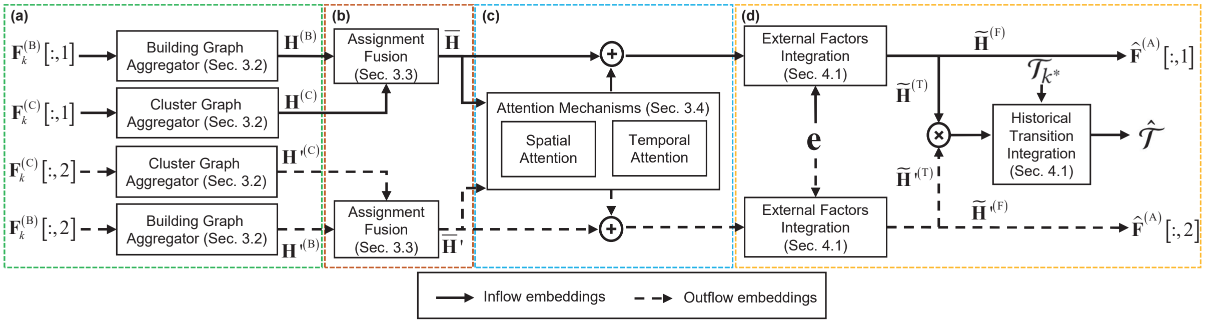

(a)
(b)
(c)
(d)
$\mathbf{F}_k^{(B)}[:,1]$
$\mathbf{F}_k^{(C)}[:,1]$
$\mathbf{F}_k^{(C)}[:,2]$
$\mathbf{F}_k^{(B)}[:,2]$
Building Graph Aggregator (Sec. 3.2)
Cluster Graph Aggregator (Sec. 3.2)
Cluster Graph Aggregator (Sec. 3.2)
Building Graph Aggregator (Sec. 3.2)
$\mathbf{H}^{(B)}$
$\mathbf{H}^{(C)}$
$\mathbf{H}'^{(C)}$
$\mathbf{H}'^{(B)}$
Assignment Fusion (Sec. 3.3)
Assignment Fusion (Sec. 3.3)
$\overline{\mathbf{H}}$
$\overline{\mathbf{H}}'$
Attention Mechanisms (Sec. 3.4)
Spatial Attention
Temporal Attention
External Factors Integration (Sec. 4.1)
External Factors Integration (Sec. 4.1)
e
$\widetilde{\mathbf{H}}^{(F)}$
$\widetilde{\mathbf{H}}^{(T)}$
$\widetilde{\mathbf{H}}'^{(T)}$
$\widetilde{\mathbf{H}}'^{(F)}$
$\mathcal{T}_{k^*}$
Historical Transition Integration (Sec. 4.1)
$\hat{\mathcal{T}}$
$\hat{\mathbf{F}}^{(A)}[:,1]$
$\hat{\mathbf{F}}^{(A)}[:,2]$
Inflow embeddings
Outflow embeddings

Fig. 8. Overview of the GAEFT's architecture, which consists of four phases.

(a) *Graph Aggregation*: GAEFT takes in the building-level flows $\mathbf{F}_k^{(B)}$ and cluster-level flows $\mathbf{F}_k^{(C)}$, where $\mathbf{F}_k^{(B)}[:,1]$ and $\mathbf{F}_k^{(C)}[:,1]$ correspond to inflows, and $\mathbf{F}_k^{(B)}[:,2]$ and $\mathbf{F}_k^{(C)}[:,2]$ correspond to outflows. Taking the *inflows* in the upper half of Fig. 8 as an example, we first use graph aggregators (Sec. 3.2) to capture the local feature correlations across the neighboring nodes (*i.e.,* buildings or clusters). Specifically, given the inputs $\mathbf{F}_k^{(B)}[:,1]$ and $\mathbf{F}_k^{(C)}[:,1]$, GAEFT generates the building-level inflow embeddings, $\mathbf{H}^{(B)}$, and cluster-level inflow embeddings, $\mathbf{H}^{(C)}$, respectively based on the *building graph aggregator* and *cluster graph aggregator*.

(b) *Assignment Fusion*: We then incorporate the $\mathbf{H}^{(\mathrm{C})}$ and $\mathbf{H}^{(\mathrm{B})}$ through *assignment fusion* (Sec. 3.3) in order to fuse the embedded features from the buildings and clusters. This way, we obtain the $\overline{\mathbf{H}}$, *i.e.,* the building-level inflow embeddings.

(c) *Spatial and Temporal Attentions*: Given $\overline{\mathbf{H}}$, we then adopt *attention mechanisms* (Sec. 3.4) in terms of *spatial* and *temporal attentions* for $\overline{\mathbf{H}}$ to further capture the local and global spatial correlations across the close and distant buildings/clusters as well as the temporal correlations between the historical and the predicted time intervals. We have designed *gated fusion* to incorporate the spatial and temporal attentions.

(d) *Integration with External Factors & Historical Transitions*: The generated outputs after the attentions will be fused with $\mathbf{e}$, the external factors such as weather conditions and weekdays/weekends (Sec. 4.1), to assist GAEFT in learning the complex crowd mobility environments. The resulting embeddings comprise two parts: $\widetilde{\mathbf{H}}^{(\mathrm{T})}$ for transition prediction, and $\widetilde{\mathbf{H}}^{(\mathrm{F})}$ for final AP-level inflow prediction ($\widehat{\mathbf{F}}^{(\mathrm{A})}[:, 1]$). GAEFT fuse the embeddings from upper and lower halves, and merge with the transitions in the same historical time interval (Sec. 4.1), denoted as $\mathcal{T}_{k^*}$, and finally generate the final building-cluster transition prediction $\widehat{\mathcal{T}}$.

The aforementioned process applies to the lower half in processing the outflows, which will lead to the predicted AP-level outflows, $\widehat{\mathbf{F}}^{(\mathrm{A})}[:, 2]$. We further discuss the details of each component in the following sections.

3.2 Graph Aggregation for Spatial Feature Learning

Considering the correlations as *edges* connecting different campus locations as *nodes*, we formulate the locations into a *graph* and introduce the graph aggregation to capture the spatial features of each location, *i.e.,* a building or a cluster. Here we will formulate the spatial characteristics of a node as the node *embeddings*, which can be generated by *aggregating* the spatial features of its neighbor nodes, *i.e.,* the sum of features of its neighbor nodes weighted by the correlations between them. As the inflows and outflows of each building or cluster encode the crowd mobility patterns in its neighborhood, we adopt the in/outflows as the node features for our formulation.

To generate the building/cluster-level inflow/outflow embeddings, we first design two graph aggregators respectively for buildings and clusters to capture the inherent spatial building-to-building and cluster-to-cluster correlations from the inputs. In practice, the features of each node are more correlated with those of its neighboring nodes than others that are far apart [36]. Therefore, we first design an aggregation strategy for feature extraction based on their mutual geospatial distances.

Specifically, we consider a building graph $\mathcal{G}^{(\mathrm{B})}$ that consists of $N^{(\mathrm{B})}$ buildings as nodes and their geospatial distances as edges. Similarly, we use the centroid points of clusters as nodes and the mutual geospatial distances between the centroid points as edges, and form the cluster graph $\mathcal{G}^{(\mathrm{C})}$. Taking building-level inflows as an example, we define $d[b, b'] = \frac{1}{g[b,b']}$ (km^{-1}), *i.e.,* the inverse geographic distance between buildings b and b'; we set $d[b, b'] = 0$ if $b = b'$. Then the aggregated flow features at the building b, denoted by $\overline{\mathbf{F}}_k^{(\mathrm{B})}[b] \in \mathbb{R}^1$, is given by the weighted sum of the inflows from other buildings, $\mathbf{F}_k^{(\mathrm{B})}[b', 1] \in \mathbb{R}^1$, *i.e.,*

$$\overline{\mathbf{F}}_k^{(\mathrm{B})}[b] = \sum_{b'}^{N^{(\mathrm{B})}} \frac{d[b, b'] \cdot \mathbf{F}_k^{(\mathrm{B})}[b', 1]}{\sum_{b'}^{N^{(\mathrm{B})}} d[b, b']}. \tag{3}$$

In other words, the nodes that are closer to each other have a higher $d[b, b']$ and subsequently stronger correlations in the resulting embeddings. Eq. (3) applies to outflows (resulting in $\overline{\mathbf{F}'}_k^{(\mathrm{B})}[b]$), and we can similarly formulate the aggregated flow features for the cluster-level flows (replacing "B" with "C").

Given above, $\overline{\mathbf{F}}_k^{(\mathrm{B})}[b]$ becomes the *aggregation* of the crowd flow features in the neighborhood of a node. Then we incorporate the original flow feature, $\mathbf{F}_k^{(\mathrm{B})}[b, 1]$, of building b for a comprehensive representation of building b's inflow embeddings. Specifically, the building-level inflow embeddings of building b, $\mathbf{H}_k^{(\mathrm{B})}[b] \in \mathbb{R}^w$, will be

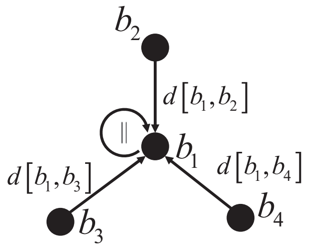


Fig. 9. Illustration of building graph aggregator. || represents concatenation.

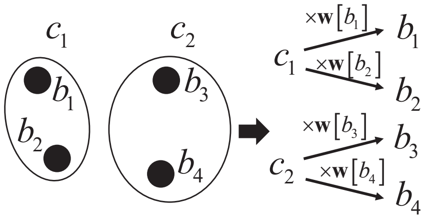


Fig. 10. Illustration of assignment fusion from clusters to buildings.

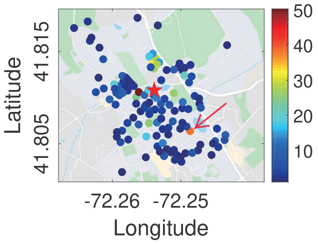


Fig. 11. Daily average number of transitions from an academic building (red star) to other buildings (2020-10-22 to 2020-11-10).

given by

$$\mathbf{H}_k^{(\mathrm{B})}[b] = \mathsf{ReLu}\left(\left(\mathbf{F}_k^{(\mathrm{B})}[b,1] \,\Big|\Big|\, \overline{\mathbf{F}}_k^{(\mathrm{B})}[b]\right)\cdot \mathbf{W}\right), \tag{4}$$

where $\mathbf{W} \in \mathbb{R}^{2\times w}$ is a learnable parameter matrix of the aggregator, w is the size of building-level inflow embeddings, and || is concatenation operation. We then use $\mathbf{H}^{(\mathrm{B})} \in \mathbb{R}^{K\times N^{(\mathrm{B})}\times w}$ to represent the inflow embeddings at all buildings during all K intervals. Similarly, we obtain $\mathbf{H'}^{(\mathrm{B})}$, $\mathbf{H}^{(\mathrm{C})}$, and $\mathbf{H'}^{(\mathrm{C})}$ as the building-level outflow embeddings, cluster-level inflow embeddings, and cluster-level outflow embeddings, respectively.

Fig. 9 illustrates an example of our aggregation strategy to extract features of building b_1 based on its correlations with other peers b_2, b_3, and b_4 (denoted as $d[b_1, b_2]$, $d[b_1, b_3]$, and $d[b_1, b_4]$). Given $\mathbf{H}_k^{(\mathrm{B})}[1]$ based on Eq. (3), we concatenate the flow features $\mathbf{F}_k^{(\mathrm{B})}[1, 1]$ at b_1 using Eq. (4), and form a comprehensive representation for the later assignment fusion.

3.3 Assignment Fusion between Embeddings of Cluster and Building

The cluster-level inflow and outflow embeddings, $\mathbf{H}^{(\mathrm{C})}$ and $\mathbf{H}'^{(\mathrm{C})}$, obtained from the graph aggregator contain the spatial features of the cluster that consists of multiple buildings, and hence have inherent correlations with the building-level inflow/outflow embeddings, $\mathbf{H}^{(\mathrm{B})}$ and $\mathbf{H'}^{(\mathrm{B})}$. We can leverage such inherent correlations to design a fusion mechanism in order to further reconstruct the crowd flows and transitions. This way, we can enhance the model learnability and alleviate the impacts from the input data sparsity.

To this end, we design an *assignment fusion* mechanism to merge the cluster-level inflow/outflow embeddings into the building-level inflow/outflow embeddings. Specifically, taking the building-level inflow embeddings as an example, we map the total $N^{(\mathrm{C})}$ cluster-level inflow embeddings, $\mathbf{H}_k^{(\mathrm{C})}$, to the the total $N^{(\mathrm{B})}$ buildings via a trainable indicator matrix $\mathbf{I}$. Here the embeddings of each cluster are assigned to the buildings only in this cluster. Then we add the mapped cluster-level inflow embeddings to the building-level inflow embeddings, $\mathbf{H}_k^{(\mathrm{B})}$, *i.e.,*

$$\overline{\mathbf{H}}_k = \mathbf{H}_k^{(\mathrm{B})} + \mathbf{I}\cdot\mathbf{H}_k^{(\mathrm{C})}, \tag{5}$$

where $\mathbf{I} \in \mathbb{R}^{N^{(\mathrm{B})}\times N^{(\mathrm{C})}}$ is a trainable indicator matrix representing which cluster a building should be mapped to, *i.e.,*

$$\mathbf{I}[b,c] = \begin{cases} \mathbf{w}[b], & \text{if a building } b \text{ should be mapped to a cluster } c; \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

where $\mathbf{w} = \left[\mathbf{w}[1], ..., \mathbf{w}[b], ..., \mathbf{w}[N^{(\mathrm{B})}]\right] \in \mathbb{R}^{N^{(\mathrm{B})}}$ represents the vector of trainable parameters. We note that the positive (negative) value of $\mathbf{w}[b]$ represents the positive (negative) correlations between building b and cluster c if the building b is in the cluster c. GAEFT's training will find the values of $\mathbf{w}$ and determine the correlations. This way, GAEFT will leverage different weights to characterize how the cluster-level and building-level inflow/outflow embeddings are correlated, and hence help identify their correlations for further model learning.

Fig. 10 illustrates an example of two clusters c_1 and c_2, and the respective buildings within each of them. Given these, our assignment fusion will aim to learn the mapping weights $\mathbf{w}$ such that the embeddings of c_1 will be only mapped to b_1 and b_2, and similarly the embeddings of c_2 will be mapped towards b_3 and b_4.

The resulting merged building-level inflow embeddings at all buildings during all time intervals, $\overline{\mathbf{H}}$, become

$$\overline{\mathbf{H}} \in \mathbb{R}^{K\times N^{(\mathrm{B})}\times w} = \left\{\overline{\mathbf{H}}_1[1], \ldots, \overline{\mathbf{H}}_k[b], \ldots, \overline{\mathbf{H}}_K[N^{(\mathrm{B})}]\right\}. \tag{7}$$

Similarly, we can obtain the merged building-level outflow embeddings, $\overline{\mathbf{H}}' \in \mathbb{R}^{K\times N^{(\mathrm{B})}\times w}$.

3.4 Spatio-Temporal Attention Mechanisms and Their Fusion

With the merged building-level inflow and outflow embeddings, $\overline{\mathbf{H}}$ and $\overline{\mathbf{H}'}$, we then add two masks in terms of spatial and temporal attentions to enhance the spatio-temporal learnability of GAEFT.

3.4.1 Spatial Attention. We note that the graph aggregator described in Eq. (3) focuses on the *local* correlations among the buildings which are close to each other in terms of geo-distances. However, there still exist crowd transitions that span across buildings which are far apart due to the specific commute routines or preferences of crowds. For instance, as illustrated in Fig. 11, we can notice the crowd transitions from an academic building to another building that is distant away. Simply using the graph aggregators may likely neglect such transitions and the inherent correlations. Therefore, it is essential to incorporate and differentiate such multi-level (local and global) mobility trends of crowd mobility to enhance the model learnability.

To this end, we propose a multi-head spatial attention mask [40] on the building-level inflow and outflow embeddings, *i.e.,* $\overline{\mathbf{H}}_k \in \mathbb{R}^{N^{(\mathrm{B})}\times w} = \left\{\overline{\mathbf{H}}_k[1], \ldots, \overline{\mathbf{H}}_k[N^{(\mathrm{B})}]\right\}$ and $\overline{\mathbf{H}}'_k \in \mathbb{R}^{N^{(\mathrm{B})}\times w} = \left\{\overline{\mathbf{H}}'_k[1], \ldots, \overline{\mathbf{H}}'_k[N^{(\mathrm{B})}]\right\}$, at each time interval k. We let Sp label the terms that are related to <u>sp</u>atial attention. For the m-th attention head ($m \in \{1, \ldots, M\}$) for the building b during each time interval k, we first obtain a *query* matrix, $\mathbf{q}_{k,m}^{(\mathrm{Sp})}$, by concatenating the building-level inflow/outflow embeddings $\overline{\mathbf{H}}_k$ and $\overline{\mathbf{H}}'_k$, *i.e.*,

$$\mathbf{q}_{k,m}^{(\mathrm{Sp})} = \mathtt{tanh}\left(\mathbf{Q}_m^{(\mathrm{Sp})} \cdot \mathtt{Flatten}\left(\overline{\mathbf{H}}_k \middle|\middle| \overline{\mathbf{H}}'_k\right)\right), \tag{8}$$

where $\mathbf{Q}_m^{(\mathrm{Sp})} \in \mathbb{R}^{w/M\times 2N^{(\mathrm{B})}w}$ is a trainable parameter matrix and $||$ is the concatenation operation. We generate the query matrices from flattened concatenation of building-level inflow/outflow embeddings, $\overline{\mathbf{H}}_k \,||\, \overline{\mathbf{H}}'_k$ to account for their similarities in terms of embedded flow patterns.

Then we find the *key* matrices, $\mathbf{x}_{k,m}^{(\mathrm{Sp})}[b]$ and $\mathbf{x}_{k,m}^{(\mathrm{Sp})'}[b]$, for each building b and each interval k, which are respectively generated by the building-level inflow and outflow embeddings, $\overline{\mathbf{H}}_k[b]$ and $\overline{\mathbf{H}}'_k[b] \in \mathbb{R}^w$. Taking building-level inflow embeddings as an example, we have

$$\mathbf{x}_{k,m}^{(\mathrm{Sp})}[b] = \mathtt{tanh}\left(\mathbf{X}_m^{(\mathrm{Sp})} \cdot \overline{\mathbf{H}}_k[b]\right), \tag{9}$$

where $\mathbf{X}_m^{(\mathrm{Sp})} \in \mathbb{R}^{w/M\times w}$ is a trainable parameter matrix.

By comparing the key matrix, $\mathbf{x}_{k,m}^{(\mathrm{Sp})}[b]$, with query matrix, $\mathbf{q}_{k,m}^{(\mathrm{Sp})}$, through an inner product, we can obtain the spatial attention weight of building-level inflow embeddings $\boldsymbol{\alpha}_{k,m}^{(\mathrm{Sp})}[b]$ to characterize the mobility trend between a building b and others, *i.e.,*

$$\boldsymbol{\alpha}_{k,m}^{(\mathrm{Sp})}[b] = \mathtt{sigmoid}\left(\frac{\mathbf{q}_{k,m}^{(\mathrm{Sp})} \cdot \left(\mathbf{x}_{k,m}^{(\mathrm{Sp})}[b]\right)^{\intercal}}{\sqrt{w/M}}\right). \tag{10}$$

The attention weights of all the buildings, $\boldsymbol{\alpha}_{k,m}^{(\mathrm{Sp})} = \left\{\boldsymbol{\alpha}_{k,m}^{(\mathrm{Sp})}[1], \ldots, \boldsymbol{\alpha}_{k,m}^{(\mathrm{Sp})}[N^{(\mathrm{B})}]\right\}$, form $\boldsymbol{\alpha}_{k,m}^{(\mathrm{Sp})} \in \mathbb{R}^{N^{(\mathrm{B})}}$, which encodes the mobility trend of the inflows. Similarly, we can obtain $\boldsymbol{\alpha}_{k,m}^{(\mathrm{Sp})'}$ for the mobility trend of the outflows.

We then generate the *value* matrices for building-level inflow embeddings, $\mathbf{v}_{k,m}^{(\mathrm{Sp})}[b]$, by

$$\mathbf{v}_{k,m}^{(\mathrm{Sp})}[b] = \mathtt{tanh}\left(\mathbf{V}_m^{(\mathrm{Sp})} \cdot \overline{\mathbf{H}}_k[b]\right), \tag{11}$$

where $\mathbf{V}_m^{(\mathrm{Sp})} \in \mathbb{R}^{w/M \times w}$ is a trainable parameter matrix (similarly for outflow). Based on Eqs. (10) and (11), we generate the output of spatial attention mechanisms for the inflow embeddings of building b at time interval k, $\mathbf{A}_k^{(\mathrm{Sp})}[b] \in \mathbb{R}^w$, by concatenating weighted value matrices for all M attention heads:

$$\mathbf{A}_k^{(\mathrm{Sp})}[b] = \Big|\Big|_{m}^{M} \boldsymbol{\alpha}_{k,m}^{(\mathrm{Sp})}[b] \cdot \mathbf{v}_{k,m}^{(\mathrm{Sp})}[b]. \tag{12}$$

We finally form the spatial attention output for inflows (similarly for outflows), denoted as $\mathbf{A}^{(\mathrm{Sp})} \in \mathbb{R}^{K \times N^{(\mathrm{B})} \times w}$, for all buildings and time intervals as, *i.e.,*

$$\mathbf{A}^{(\mathrm{Sp})} = \left\{\mathbf{A}_1^{(\mathrm{Sp})}[1], \ldots, \mathbf{A}_k^{(\mathrm{Sp})}[b], \ldots, \mathbf{A}_K^{(\mathrm{Sp})}[N^{(\mathrm{B})}]\right\}. \tag{13}$$

3.4.2 Temporal Attention. To further capture the temporal correlations between the crowd flows at different time intervals, we design a multi-head temporal attention mechanism. Let Tp be the symbol indicating the terms within temporal attention. For the p-th output of the total P attention heads for historical time interval k at building b, we generate the query matrix, $\mathbf{q}_{b,p}^{(\mathrm{Tp})}$, by concatenating the inflow/outflow embeddings of a building b for all historical K time intervals, $\overline{\mathbf{H}}[b] \in \mathbb{R}^{K \times w} = \left\{\overline{\mathbf{H}}_1[b], \ldots, \overline{\mathbf{H}}_k[b]\right\}$ and $\overline{\mathbf{H}}'[b] \in \mathbb{R}^{K \times w} = \left\{\overline{\mathbf{H}}_1'[b], \ldots, \overline{\mathbf{H}}_k'[b]\right\}$, *i.e.,*

$$\mathbf{q}_{b,p}^{(\mathrm{Tp})} = \mathtt{tanh}\left(\mathbf{Q}_p^{(\mathrm{Tp})} \cdot \mathtt{Flatten}\left(\overline{\mathbf{H}}[b] \Big|\Big| \overline{\mathbf{H}}'[b]\right)\right), \tag{14}$$

where $\mathbf{Q}_p^{(\mathrm{Tp})} \in \mathbb{R}^{w/P \times 2Kw}$ is a trainable parameter matrix.

We focus on temporal attention for the building-level inflow embeddings as follows. Following the same manner as spatial attention mechanism, we generate the key matrices $\mathbf{x}_{k,p}^{(\mathrm{Tp})}[b]$, attention weights $\boldsymbol{\alpha}_{k,p}^{(\mathrm{Tp})}[b]$, and value matrices $\mathbf{v}_{k,p}^{(\mathrm{Tp})}[b]$ by

$$\mathbf{x}_{k,p}^{(\mathrm{Tp})}[b] = \mathtt{tanh}\left(\mathbf{X}_p^{(\mathrm{Tp})} \cdot \overline{\mathbf{H}}_k[b]\right), \boldsymbol{\alpha}_{k,p}^{(\mathrm{Tp})}[b] = \mathtt{sigmoid}\left(\frac{\mathbf{q}_{b,p}^{(\mathrm{Tp})} \cdot \left(\mathbf{x}_{k,p}^{(\mathrm{Tp})}[b]\right)^{\intercal}}{\sqrt{w/P}}\right), \mathbf{v}_{k,p}^{(\mathrm{Tp})}[b] = \mathtt{tanh}\left(\mathbf{V}_p^{(\mathrm{Tp})} \cdot \overline{\mathbf{H}}_k[b]\right), \tag{15}$$

where $\mathbf{X}_p^{(\mathrm{Tp})} \in \mathbb{R}^{w/P \times w}$ and $\mathbf{V}_p^{(\mathrm{Tp})} \in \mathbb{R}^{w/P \times w}$ are the corresponding learnable parameter matrices. The attention weights, $\boldsymbol{\alpha}_{k,p}^{(\mathrm{Tp})}$, represent the importance of the building-level inflow embeddings of time interval k in all K historical time intervals, and the higher values represent the higher correlation between time interval k and the future time interval. Then with the concatenation of weighted value matrices for all P heads, the temporal attention output for the inflow embeddings of building b, $\mathbf{A}_k^{(\mathrm{Tp})}[b] \in \mathbb{R}^w$, becomes

$$\mathbf{A}_k^{(\mathrm{Tp})}[b] = \Big\|_{p}^{P} \boldsymbol{\alpha}_{k,p}^{(\mathrm{Tp})}[b] \cdot \mathbf{v}_{k,p}^{(\mathrm{Tp})}[b]. \tag{16}$$

We finally have the output, denoted as $\mathbf{A}^{(\mathrm{Tp})} \in \mathbb{R}^{K \times N^{(\mathrm{B})} \times w}$, for all buildings at all time intervals,

$$\mathbf{A}^{(\mathrm{Tp})} = \left\{ \mathbf{A}_1^{(\mathrm{Tp})}[1], \ldots, \mathbf{A}_k^{(\mathrm{Tp})}[b], \ldots, \mathbf{A}_K^{(\mathrm{Tp})}[N^{(\mathrm{B})}] \right\}. \tag{17}$$

3.4.3 Gated Fusion. Given the outputs of the spatial and temporal attentions for building-level inflow embeddings, *i.e.,* $\mathbf{A}^{(\mathrm{Sp})} \in \mathbb{R}^{K \times N^{(\mathrm{B})} \times w}$ and $\mathbf{A}^{(\mathrm{Tp})} \in \mathbb{R}^{K \times N^{(\mathrm{B})} \times w}$, we further fuse and add them into the previous building-level inflow embeddings, $\overline{\mathbf{H}} \in \mathbb{R}^{K \times N^{(\mathrm{B})} \times w}$ in Eq. (7). Note that the resulting spatial and temporal attentions may have different weights for different buildings and different time intervals. Therefore, we employ a *gated fusion* mechanism to further fuse the outputs from the two attention mechanisms and obtain the merged attention output $\mathbf{A}^{(\mathrm{S})} \in \mathbb{R}^{K \times N^{(\mathrm{B})} \times w}$, *i.e.,*

$$\mathbf{A}^{(\mathrm{S})} = \mathbf{S} \circ \mathbf{A}^{(\mathrm{Sp})} + (\mathbf{1} - \mathbf{S}) \circ \mathbf{A}^{(\mathrm{Tp})}, \tag{18}$$

where $\circ$ represents the element-wise product operation, and $\mathbf{1} \in \mathbb{R}^{K \times N^{(\mathrm{B})} \times w}$ is a matrix where all entries are 1. Here $\mathbf{S} \in \mathbb{R}^{K \times N^{(\mathrm{B})} \times w}$ is the weight matrix representing the relative importance of a spatial attention mask for the inflow/outflow embeddings of each building at each time interval, *i.e.,*

$$\mathbf{S} = \mathtt{sigmoid}\left(\mathbf{A}^{(\mathrm{Sp})} \cdot \mathbf{W}^{(\mathrm{Sp})} + \mathbf{A}^{(\mathrm{Tp})} \cdot \mathbf{W}^{(\mathrm{Tp})} + \mathbf{b}^{(\mathrm{S})} \right), \tag{19}$$

where matrices $\mathbf{W}^{(\mathrm{Sp})} \in \mathbb{R}^{w \times w}$, $\mathbf{W}^{(\mathrm{Tp})} \in \mathbb{R}^{w \times w}$, and $\mathbf{b}^{(\mathrm{S})} \in \mathbb{R}^w$ are learnable parameter matrices. Here we adopt a `sigmoid` activation function to enforce the value of the weight matrix to fall within the range of $[0, 1]$.

We then merge the $\mathbf{A}^{(\mathrm{S})}$ with the building-level inflow embeddings $\overline{\mathbf{H}}$ that is returned from Eq. (7), and obtain the building-level inflow embeddings at the future time interval, $\overline{\mathbf{A}} \in \mathbb{R}^{N^{(\mathrm{B})} \times w}$, through a `Dense` layer, *i.e.,*

$$\overline{\mathbf{A}} = \mathtt{Dense}\left(\overline{\mathbf{H}} + \mathbf{A}^{(\mathrm{S})} \right). \tag{20}$$

4 MODEL INTEGRATION & MULTI-TASK LEARNING

Given the building-level inflow/outflow embeddings, $\overline{\mathbf{A}}$ and $\overline{\mathbf{A}}'$ from above, we integrate external factors and historical transitions in Sec. 4.1. Then we detail the final predictions of building-cluster transitions and AP-level flows as well as the multi-task learning modeling in Sec. 4.2.

4.1 Integrating External Factors & Historical Transitions

4.1.1 External Factors. As illustrated in Fig. 12, given the vector $\mathbf{e} \in \mathbb{R}^{l_e}$ formed by l_e external factors in total, we first adopt two consecutive `Dense` layers followed by the `tanh` activation function to generate the hidden state of the external factors, denoted as $\mathbf{h}^{(\mathrm{e})} \in \mathbb{R}^{l'_e}$, *i.e.,*

$$\mathbf{h}^{(\mathrm{e})} = \mathtt{tanh}\left(\mathtt{Dense}(\mathtt{Dense}(\mathbf{e})) \right). \tag{21}$$

Then we will reshape the output $\mathbf{h}^{(\mathrm{e})}$ respectively towards the transitions and flows. Specifically, for transition prediction, the output dimension of Eq. (21) is set $l'_e = N^{(\mathrm{B})} \cdot w$, and after the reshape operation we will have the 2-D tensor $\mathbf{h}^{(\mathrm{T})} \in \mathbb{R}^{N^{(\mathrm{B})} \times w}$. Recall that we have the building-level inflow embeddings $\overline{\mathbf{A}}$ obtained from Eq. (20), and we will fuse $\overline{\mathbf{A}}$ with $\mathbf{h}^{(\mathrm{T})}$, *i.e.,* $\widetilde{\mathbf{H}}^{(\mathrm{T})} \in \mathbb{R}^{N^{(\mathrm{B})} \times w} = \overline{\mathbf{A}} + \mathbf{h}^{(\mathrm{T})}$, and obtain the final building-level inflow embeddiings $\widetilde{\mathbf{H}}^{(\mathrm{T})}$ for transition prediction.

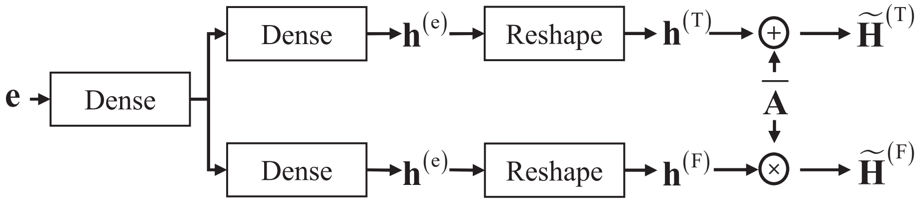


Fig. 12. Illustration of incorporating external factors $\mathbf{e}$ into building-level inflow embeddings $\overline{\mathbf{A}}$.

Similarly, for the flow prediction, we set $l'_e = w^2$ for ease of the reshape operation, and multiply $\overline{\mathbf{A}}$ with $\mathbf{h}^{(F)}$ to obtain the final building-level inflow embeddings $\widetilde{\mathbf{H}}^{(F)} \in \mathbb{R}^{N^{(B)} \times w}$ for flow prediction. These designs apply the same for the building-level outflow embeddings and subsequent predictions ($\widetilde{\mathbf{H}}'^{(T)}$ and $\widetilde{\mathbf{H}}'^{(F)}$).

4.1.2 Historical Transitions. We also observe from our data analytics that the number of transitions at a time interval k is highly similar to those transitions in the same historical time intervals (*e.g.,* the same 8:00–9:00am on the day before). Therefore, we incorporate building-cluster transitions at the historical time interval, $\mathcal{T}_{k^*} \in \mathbb{R}^{N^{(B)} \times N^{(C)} \times 2}$, in our model design (as illustrated in Fig. 8), where k^* is set as the same time interval as the future time interval on one day ago. We adopt 3 layers of 2D convolutional neural network (Conv2D) to extract the hidden feature of historical building-cluster transitions at time interval k^*, denoted by $\mathbf{H}_{k^*}^{(\mathrm{His})} \in \mathbb{R}^{N^{(B)} \times N^{(C)} \times 2}$. Specifically, we have

$$\mathbf{H}_{k^*}^{(\mathrm{His})} = \mathsf{ReLu}\left(\mathsf{Conv2D}\left(\mathsf{Conv2D}\left(\mathsf{Conv2D}\left(\mathcal{T}_{k^*}\right)\right)\right)\right), \tag{22}$$

where we adopt the zero padding for each convolutional layer and set the output channel size for the last layer as 2. We use the ReLu activation function to ensure the non-negative output values. Then the extracted hidden features, $\mathbf{H}_{k^*}^{(\mathrm{His})}$, are fused into the transition matrix generated by the building-level inflow/outflow embeddings to return the final prediction (elaborated in Sec. 4.2).

4.2 Predicting Building-Cluster Transitions and AP-level Flows

4.2.1 Building-Cluster Transitions. Recall that we obtain the building-level inflow/outflow embeddings, $\widetilde{\mathbf{H}}^{(T)} \in \mathbb{R}^{N^{(B)} \times w}$ and $\widetilde{\mathbf{H}}'^{(T)} \in \mathbb{R}^{N^{(B)} \times w}$, based on the integration of external factors for the transition prediction. We first find the sum of the $\widetilde{\mathbf{H}}^{(T)}$ and $\widetilde{\mathbf{H}}'^{(T)}$ of buildings that are in the same cluster to obtain the cluster-level inflow and outflow embeddings, $\widetilde{\mathbf{H}}^{(C)} \in \mathbb{R}^{N^{(C)} \times w}$ and $\widetilde{\mathbf{H}}^{(C)'} \in \mathbb{R}^{N^{(C)} \times w}$, respectively, *i.e.,*

$$\widetilde{\mathbf{H}}^{(C)}[c] = \sum_{b \subseteq c} \widetilde{\mathbf{H}}^{(T)}[b], \quad \widetilde{\mathbf{H}}'^{(C)}[c] = \sum_{b \subseteq c} \widetilde{\mathbf{H}}'^{(T)}[b]. \tag{23}$$

Our goal is to obtain the transitions based on the embeddings related to flows, and inspired by the matrix decomposition [19], we consider the multiplication operation to realize this. Specifically, the transitions from buildings to clusters, $\widetilde{\mathcal{T}}[:,:,1] \in \mathbb{R}^{N^{(B)} \times N^{(C)}}$, are calculated as multiplication of the building-level outflow embeddings $\widetilde{\mathbf{H}}'^{(T)}$ and cluster-level inflow embeddings $\widetilde{\mathbf{H}}^{(C)}$, *i.e.,*

$$\widetilde{\mathcal{T}}[:,:,1] = \mathsf{ReLu}\left(\widetilde{\mathbf{H}}'^{(T)} \cdot \left(\widetilde{\mathbf{H}}^{(C)}\right)^{\intercal}\right), \tag{24}$$

where we apply a ReLu activation function to ensure the prediction is always positive. Similarly, the transitions from clusters to buildings, $\widetilde{\mathcal{T}}[:,:,2]$, are given by the multiplication of the building-level inflow embeddings $\widetilde{\mathbf{H}}^{(T)}$ and cluster-level outflow embeddings $\widetilde{\mathbf{H}}'^{(C)}$, *i.e.,*

$$\widetilde{\mathcal{T}}[:,:,2] = \mathsf{ReLu}\left(\widetilde{\mathbf{H}}^{(T)} \cdot \left(\widetilde{\mathbf{H}}'^{(C)}\right)^{\intercal}\right). \tag{25}$$

We finally incorporate the hidden features of historical transitions, $\mathbf{H}_{k^*}^{(\text{His})}$, into $\widetilde{\mathcal{T}}$ by a gated fusion mechanism described in Sec. 3.4.3, and obtain the final building-cluster transition prediction, $\widehat{\mathcal{T}} \in \mathbb{R}^{N^{(\text{B})} \times N^{(\text{C})} \times 2}$.

4.2.2 AP-level Flows. Similarly, we use the building-level embeddings from the external factor fusion ($\widetilde{\mathbf{H}}^{(\text{F})}$ and $\widetilde{\mathbf{H}}'^{(\text{F})}$) to predict the AP-level flows. Specifically, we map the building-level inflow embeddings $\widetilde{\mathbf{H}}^{(\text{F})}$ to all the APs which are situated within the same building via a trainable indicator matrix, $\mathbf{J} \in \mathbb{R}^{N^{(\text{A})} \times N^{(\text{B})}}$, to generate the AP-level inflow embeddings, $\widetilde{\mathbf{F}}^{(\text{A})} \in \mathbb{R}^{N^{(\text{A})} \times w}$, *i.e.,*

$$\widetilde{\mathbf{F}}^{(\text{A})} = \mathbf{J} \cdot \widetilde{\mathbf{H}}^{(\text{F})}. \tag{26}$$

Note that $\mathbf{J}$ indicates which building an AP is located at, *i.e.*,

$$\mathbf{J}[a, b] = \begin{cases} \mathbf{u}[a], & \text{for } a \subseteq b; \\ 0, & \text{otherwise}, \end{cases} \tag{27}$$

where $\mathbf{u} = \left[\mathbf{u}[1], ..., \mathbf{u}[a], ...\mathbf{u}[N^{(\text{A})}]\right] \in \mathbb{R}^{N^{(\text{A})}}$ is the vector of the trainable parameters which are learned by the model during training. We similarly obtain the AP-level outflow embeddings, $\widetilde{\mathbf{F}}'^{(\text{A})}$. We then adopt a `Dense` layer with a `ReLu` activation function which is to ensure non-negative outputs to generate the final predictions of AP-level flows at the future time interval, $\widehat{\mathbf{F}}^{(\text{A})} \in \mathbb{R}^{N^{(\text{A})} \times 2}$ ($\widehat{\mathbf{F}}^{(\text{A})}[:, 1]$ denotes inflows and $\widehat{\mathbf{F}}^{(\text{A})}[:, 2]$ denotes outflows), *i.e.,*

$$\widehat{\mathbf{F}}^{(\text{A})}[:, 1] = \texttt{ReLu}\left(\texttt{Dense}\left(\widetilde{\mathbf{F}}^{(\text{A})}\right)\right), \quad \widehat{\mathbf{F}}^{(\text{A})}[:, 2] = \texttt{ReLu}\left(\texttt{Dense}\left(\widetilde{\mathbf{F}}'^{(\text{A})}\right)\right). \tag{28}$$

4.2.3 Multi-Task Model Training. In training `GAEFT`, we adopt the Frobenius Norms (FNs) to measure the training loss on the predicted building-cluster transitions, *i.e.,*

$$\mathcal{L}^{(\text{T})} = \sqrt{\sum_{i=1}^{2} \sum_{b}^{N^{(\text{B})}} \sum_{c}^{N^{(\text{C})}} \left(\widehat{\mathcal{T}}[b, c, i] - \mathcal{T}[b, c, i]\right)^2}, \tag{29}$$

and use the Mean Squared Errors (MSEs) to measure the loss in the predicted AP-level crowd flows, *i.e.,*

$$\mathcal{L}^{(\text{F})} = \frac{1}{2N^{(\text{A})}} \sum_{i=1}^{2} \sum_{a}^{N^{(\text{A})}} \left(\widehat{\mathbf{F}}^{(\text{A})}[a, i] - \mathbf{F}^{(\text{A})}[a, i]\right)^2. \tag{30}$$

Given above, `GAEFT` formulates a *joint loss function* for the multi-task learning, and *jointly* minimizes the weighted sum of $\mathcal{L}^{(\text{T})}$ and $\mathcal{L}^{(\text{F})}$, *i.e.,*

$$\mathcal{L} = \lambda^{(\text{T})} \cdot \mathcal{L}^{(\text{T})} + \lambda^{(\text{F})} \cdot \mathcal{L}^{(\text{F})}, \quad \lambda^{(\text{T})} \in [0, 1], \quad \lambda^{(\text{F})} \in [0, 1]. \tag{31}$$

Through the above formulation, we can leverage the inherent correlations across the crowd flows and transitions, and jointly enhance the `GAEFT`'s learnability upon the sparse mobility data.

5 EXPERIMENTAL STUDIES

We first present the experimental setup in Sec. 5.1 and then provide the experimental results in Sec. 5.2.

5.1 Experimental Evaluation Setup

5.1.1 Baselines. We compare `GAEFT` with the following four categories of baseline approaches.

(a) *Traditional Time Series Approaches*:

★ `GP`: which leverages Gaussian Process (GP) with the radial basis function (RBF) kernel.

★ `ARIMA`: which leverages the Autoregressive Integrated Moving Average (ARIMA).

(b) *Temporal Sequence Learning Approaches*:

- ⋆ Recurrent Neural Network/Long Short-Term Memory Network/Gated Recurrent Unit (RNN/LSTM/GRU): We set the dimension of each hidden state as 16 for each of them to predict the time series.
- ⋆ TPA-LSTM: which leverages LSTM with temporal attention to forecast the multivariate time series [34]. We set the size of LSTM hidden states as 128.

(c) *Spatial Learning Approaches*:

- ⋆ CNN: which uses 7-layer convolutional neural network (CNN) to predict the crowd mobility.
- ⋆ GCNN: Graph Convolutional Neural Network (GCNN) [24] considers the adjacency matrix as trainable matrices and historical values as the node features to generate the building-level inflow/outflow embeddings.

(d) *Spatio-Temporal Learning Approaches*:

- ⋆ ResNet: which takes the building-cluster transitions matrix as the input and adopts three blocks of spatio-temporal residual neural networks [48] which capture the short-, mid-, and long-term mobility patterns.
- ⋆ GEML: which leverages the grid-embedding-based multi-task learning [42] to generate the building-level inflow/outflow embeddings.
- ⋆ LGNN: which leverages the line graph neural network by fusion of the link graph convolution, node graph convolution and historical transition matrices [43].
- ⋆ CSTN: which leverages the convolution embedded LSTM-based method to form a contextualized spatial-temporal network [27].

5.1.2 Experimental Settings. For all the schemes, we leverage the crowd mobility data in the past 24 hours to predict the building-cluster transitions and AP-level flows in the future 1 hour. We evaluate the model performance based on the Wi-Fi association data from 2020-10-11 to 2020-11-10 (Fall semester), and from 2021-02-02 to 2021-04-10 (Spring semester). In our experiment studies, we mitigate the potential ping-pong effects [20] by removing the dissociation and reassociation within 30 min (since our time interval is set as 1h). The training data include the first 26 days of Fall, and the first 53 days of data for Spring, and we leave the rest of the data for model testing. We train the affinity propagation algorithm for 200 iterations and obtain 21 clusters as shown in Fig. 6. We process the data and conduct model training on a server with 1× AMD Threadripper 3960X 24-Core CPU, 128 GB RAM, and 4× Nvidia GeForce RTX3090. Our model is built on TensorFlow 2.4.0 with CUDA 11.1.

Unless otherwise stated, we use the following parameters in GAEFT by default. We set $\beta = 1$ in Eq. (2) in the spatio-temporal clustering. In affinity propagation clustering we set the diagonal of the similarity matrix to be the median of the other entries, and we set the damping factor as 0.9. We set the size of building-level inflow/outflow embeddings $w = 12$, and use $M = 1$ spatial attention heads and $P = 3$ temporal attention heads. We adopt Conv2Ds with respective channel sizes of 16, 8, and 2 for the 3 convolutional layers to encode the historical transitions as discussed in Sec. 4.1. We set $\lambda^{(\mathrm{T})} = 1/16$ and $\lambda^{(\mathrm{F})} = 15/16$ for the multi-task objective function (Eq. (31)). We set the batch size to be 64. We apply L2 regularization upon the trainable parameter matrices in the attention mechanisms (Eqs. (8), (9), (11), (14), and (15)), *i.e.,* $\mathbf{Q}_m^{(\mathrm{Sp}}$, $\mathbf{X}_m^{(\mathrm{Sp})}$, $\mathbf{V}_m^{(\mathrm{Sp})}$, $\mathbf{Q}_p^{(\mathrm{Tp})}$, $\mathbf{X}_p^{(\mathrm{Tp})}$, and $\mathbf{V}_p^{(\mathrm{Tp})}$, where the regularization parameters are set to 0.05. We add a Dropout layer to the building-level inflow/outflow embeddings (Eq. (7)) which are the inputs of the attention module with a dropout rate of 0.1. The model is trained for 5,000 iterations with a learning rate of 0.005 by Adam optimizer. The training time is 26min based on our default parameter settings. The average inference time for each target time interval is 4.47ms based on our test data.

We use the following metrics for the performance evaluation. Given the predicted building-cluster transitions $\widehat{\mathcal{T}_r}$ and the ground truths $\mathcal{T}_r$ at time interval r out of totally R intervals in the testing set, we define the Average

Table 2. Overall performance comparison.

Schemes	Transitions		Flows	
	FN	ARSE$^{(\mathrm{T})}$	MSE	ARSE$^{(\mathrm{F})}$
GP	187.973	2.564	19.726	3.779
ARIMA	198.427	2.706	22.381	3.915
RNN	133.607	1.822	7.488	2.535
GRU	115.747	1.579	4.855	2.091
LSTM	110.314	1.505	4.207	1.946
TPA-LSTM	108.205	1.476	7.157	2.470
CNN	139.830	1.907	12.959	3.165
GCNN	96.294	1.313	6.280	2.396
ResNet	98.313	1.341	5.098	2.172
GEML	104.107	1.420	3.717	1.832
LGNN	95.793	1.306	7.099	2.386
CSTN	87.145	1.189	7.025	2.414
GAEFT	**84.603**	**1.154**	**3.173**	**1.697**

Table 3. Weekday/weekend predictions.

Schemes	Transitions				Flows			
	Weekdays		Weekends		Weekdays		Weekends	
	FN	ARSE$^{(\mathrm{T})}$	FN	ARSE$^{(\mathrm{T})}$	MSE	ARSE$^{(\mathrm{F})}$	MSE	ARSE$^{(\mathrm{F})}$
GP	199.007	2.714	167.481	2.284	22.087	3.947	15.341	3.466
ARIMA	208.831	2.848	179.104	2.443	25.600	4.120	16.402	3.536
RNN	138.993	1.896	123.606	1.686	7.849	2.599	6.819	2.415
GRU	118.736	1.619	110.197	1.503	4.752	2.067	5.047	2.134
LSTM	112.678	1.537	105.923	1.445	4.153	1.938	4.306	1.961
TPA-LSTM	110.041	1.501	104.795	1.429	7.380	2.516	6.743	2.384
CNN	144.966	1.977	130.293	1.777	13.735	3.270	11.519	2.969
GCNN	95.650	1.305	97.491	1.330	6.501	2.443	5.871	2.309
ResNet	98.450	1.343	98.058	1.337	5.044	2.162	5.197	2.191
GEML	106.148	1.448	100.317	1.368	3.687	1.830	3.773	1.834
LGNN	96.018	1.310	95.375	1.301	7.285	2.418	6.753	2.326
CSTN	88.576	1.208	84.487	1.152	7.306	2.470	6.503	2.310
GAEFT	**84.755**	**1.156**	**84.322**	**1.150**	**2.981**	**1.656**	**3.530**	**1.772**

Root Squared Error (ARSE$^{(\mathrm{T})}$) as

$$\mathrm{ARSE}^{(\mathrm{T})} = \frac{1}{R}\sum_{r=1}^{R}\sqrt{\frac{1}{2N^{(\mathrm{B})}N^{(\mathrm{C})}}\sum_{i=1}^{2}\sum_{b=1}^{N^{(\mathrm{B})}}\sum_{c=1}^{N^{(\mathrm{C})}}\left(\widehat{\mathcal{T}}_r[b,c,i]-\mathcal{T}_r[b,c,i]\right)^2}. \tag{32}$$

We evaluate the prediction of building-cluster transitions by ARSE$^{(\mathrm{T})}$ and Frobenius Norms (denoted as FNs; same as Eq. (29)). Note that FNs of transitions represent the square root of sum of differences between predictions and ground-truths over all building-cluster pairs.

For prediction of flows of APs, the corresponding ARSE$^{(\mathrm{F})}$ is formulated by

$$\mathrm{ARSE}^{(\mathrm{F})} = \frac{1}{R}\sum_{r=1}^{R}\sqrt{\frac{1}{2N^{(\mathrm{A})}}\sum_{i=1}^{2}\sum_{a=1}^{N^{(\mathrm{A})}}\left(\widehat{\mathbf{F}}_r^{(\mathrm{A})}[a,i]-\mathbf{F}_r^{(\mathrm{A})}[a,i]\right)^2}. \tag{33}$$

We then adopt ARSE$^{(\mathrm{F})}$ and MSEs (same as Eq. (30)) for AP-level flow prediction.

5.2 Experimental Results

5.2.1 Overall Model Performance. We compare the model performance of different algorithms in Table 2. In summary, GAEFT has achieved overall lower errors than the baseline models regarding transitions (by 26.6% reduction on average) and flows (by 41.7% reduction on average). Regarding each of the four baseline categories, *i.e.,* traditional time series approaches, temporal sequence learning, spatial learning, and spatio-temporal learning, GAEFT improves the accuracy by 56.2%, 27.2%, 25.8%, and 11.8%, respectively, on average for transition prediction and by 70.4%, 33.6%, 50.1%, and 31.3%, respectively, on average for AP-level flow prediction.

Specifically, traditional machine learning models such as GP and ARIMA, might not fully capture the spatial correlations between locations as well as the spatio-temporal features at their neighborhood. Temporal sequence learning approaches like RNN, LSTM, GRU and TPA-LSTM are only able to capture the temporal correlations of the inflow/outflow features of each building. Regarding the spatial learning approaches, the convolution operations in CNN focus on the building-cluster transition matrices. GCNN encodes the spatial correlations between buildings into the trainable adjacency matrix of graph convolutional network, which might underestimate the temporal correlations across historical time intervals. From the results, we can see solely using the temporal sequence learning or spatial learning cannot fully capture the complex and sparse crowd mobility on campus.

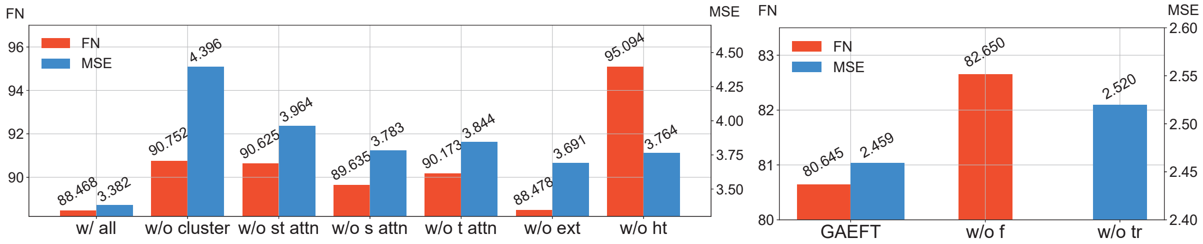
In terms of spatio-temporal learning, ResNet cannot fully capture the features of the highly sparse building-to-cluster transition matrices. GEML takes in both geo-distance and connectivity between buildings as the building correlations for the graph aggregation operation. However, its formulation might discard the aggregated spatial features of the inflows and outflows in the LSTM structure. LGNN captures the spatio-temporal correlations between buildings and clusters through the link and node graphs, encoding the information of historical transitions. However, LGNN does not consider differentiating the contributions of different building and temporal features. CSTN takes the spatial and temporal characteristics of transition patterns and the impact of external factors including weather and weekdays/weekends into account. We note that GAEFT shows slight improvements compared to CSTN regarding the transition prediction. This is likely because CSTN might also account for individual contributions of different buildings to the transitions and yield good accuracy. Nevertheless, unlike GAEFT, CSTN might not effectively take into account the spatio-temporal correlations of the building neighborhoods encoded in the cluster-level inflow/outflow embeddings. In addition, CSTN might not fully capture the building features essential for AP-level crowd flow modeling, leading to degradation in the flow prediction accuracy.

Different from the above models, GAEFT captures the nearby spatial correlations via the novel graph aggregator. In addition, the spatial and temporal attention mechanisms jointly learn the spatial and temporal correlations in the whole building network. The extraction of building-level inflow/outflow embeddings is further enhanced by incorporation of the spatio-temporal knowledge of building clusters. Therefore, GAEFT outperforms the above state-of-the-art methods.

5.2.2 Impacts of Weekdays/Weekends. We compare GAEFT with other baselines and state-of-the-arts during weekdays and weekends in Table 3, from which we can see that GAEFT outperforms other schemes in terms of accuracy and robustness. For most of the models studied, the accuracy on weekends is generally higher than that on weekdays for both building-cluster transitions and AP-level flows. This is probably due to more activities on campus on weekdays than those on weekends, leading to more complex mobility patterns for mobility modeling. We also note that since the building-cluster transitions and AP-level flows exhibit different patterns on weekends and weekdays (Sec. 2.2), incorporating external factors such as indicators of weekdays can help maintain accuracies of GAEFT.

5.2.3 Ablation Studies. We conduct the ablation studies regarding multiple design components in GAEFT.

(a) *Importance of Major Components of* GAEFT: We first study in Fig. 13a the impacts of the five major components in GAEFT upon the transitions (using FNs) and flows (using MSEs). We compare GAEFT (denoted as w/ all) with its variants that remove each of the components: *(i)* cluster-level inflow/outflow embeddings (w/o cluster), *(ii)* both spatial and temporal attentions (w/o st attn), *(iii)* spatial attention (w/o s attn), *(iv)* temporal attention (w/o t attn), *(v)* external factors (w/o ext), and *(vi)* historical transitions (w/o ht). The results are based on the last 7 days' data of the 53 days' data in the training set of 2021 Spring.



(a) Ablation study on major components of GAEFT.

(b) Effectiveness of multi-task learning.

Fig. 13. Ablation studies on: (a) major components of GAEFT; (b) multi-task learning.

We can see that all the five components help improve the model performance. In terms of transition predictions (FNs), we can observe that historical transitions (Sec. 4.1), spatio-temporal clustering (Sec. 3.2), and spatio-temporal attentions (Sec. 3.4) contribute the most. The historical transitions and clustered buildings help mitigate the sparsity within the crowd mobility data, while spatio-temporal attentions particularly improves the learnability upon the global correlations among the clusters and buildings. In terms of flow predictions (MSEs), we can see that spatio-temporal clustering (Sec. 3.2) and the spatial and temporal attentions (Sec. 3.4) lead to the most significant improvements. Temporal attention provide slightly more improvements in transitions and flows than the spatial attention due to the more complex temporal dynamics in crowd flows and transitions.

(b) *Effectiveness of Multi-task Learning*: We further investigate the effectiveness of mulit-task learning of GAEFT by training and testing the model with only building-cluster transition prediction (denoted as "w/o f", *i.e.,* $\lambda^{(\mathrm{F})} = 0$) and with only AP-level flow prediction (denoted as "w/o tr", *i.e.,* $\lambda^{(\mathrm{T})} = 0$). The results are based on the last 5 days' data of the 26 days' data in the training set of 2020 Fall. Fig. 13b shows that the multi-task learning enhances the predictions regarding the crowd flows and transitions. The inflows/outflows are aggregated from, and hence are inherently correlated with, all the transitions arriving at/departing from each locations. Our GAEFT incorporates such correlations and, therefore, improves the prediction accuracy.

5.2.4 *Sensitivity Studies.*

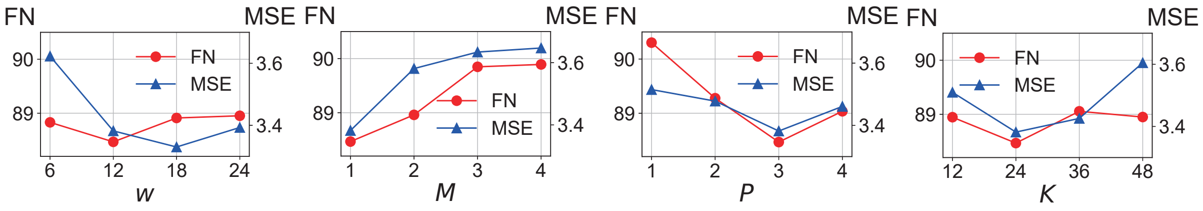
We conduct sensitivity studies regarding four important parameters in Fig. 14, based on the last 7 days' data of the 53 days' data in the training set of 2021 Spring.

Fig. 14a shows the FNs and MSEs versus the size of building-level inflow/outflow embeddings in the graph aggregator, w ($w \in \{6, 12, 18, 24\}$). When the embedding size is small (say, 6), it may not be able to capture enough mobility information and hence low accuracy is achieved. As w increases, the accuracy of GAEFT improves. However, as w further increases, noisy information may be captured and degrade the performance. Based on above, we set $w = 12$ to maintain an overall high accuracy in our multi-task learning.

Fig. 14b shows the FNs and MSEs versus number of spatial attention heads, M ($M \in \{1, 2, 3, 4\}$). We can see that the prediction errors increase as M increases mainly due to the more noisy spatial mobility information captured by the additional but redundant attention heads. Therefore, we set $M = 1$ by default.

Fig. 14c further shows the prediction errors given different numbers of temporal attention heads, *i.e.,* $P \in \{1, 2, 3, 4\}$. As P increases, GAEFT is able to capture more information and differentiate the embeddings between different time intervals, leading to the performance improvements. However, as P further increases, more noisy features from the crowd mobility data might be learned, and hence the errors of GAEFT increase, showing a diminishing return. Therefore, we set $P = 3$ by default.

Besides above, we evaluate the impact of number of historical intervals (denoted as K) in Fig. 14d. We can observe that GAEFT might achieve large errors given a time window which is either too short or too long. When $K = 24$ time intervals, the historical time intervals may provide adequate information for the model learning.



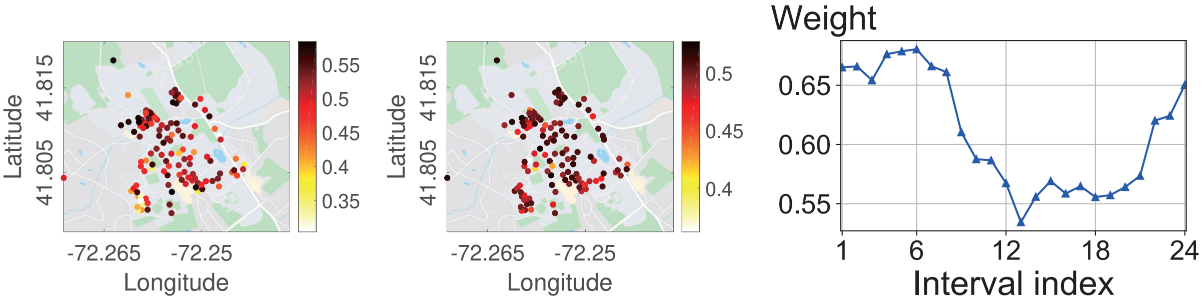
(a) Embedding size. (b) # Spatial attention heads. (c) # Temporal attention heads. (d) Length of time intervals.

Fig. 14. Sensitivity study of (a) size of building-level inflow/outflow embeddings (w); (b) the number of spatial attention heads (M); (c) the number of temporal attention heads (P); and (d) the length of time intervals (K).

5.2.5 *Visualization.*

We further visualize the evaluation results of GAEFT.

(a) *Spatial and Temporal Attention Weights*: We visualize the weights of the spatial attention head for outflow embedding of all buildings, *i.e.,* $\boldsymbol{\alpha}_{k,1}^{(\mathrm{Sp})}$ (Eq. (10) in Sec. 3.4), at 11am–12pm (Fig. 15a) and 0am–1am (Fig. 15b), respectively, on the same Friday. The spatial attention mechanism differentiates the spatial features of buildings based on the their mobility patterns. The higher $\boldsymbol{\alpha}_{k,1}^{(\mathrm{Sp})}[b]$ indicates larger contribution of features of building b to the final prediction. We can see that the differences in the weights between buildings vary with time, and are more significant at daytime than at midnight. This is mainly because the number of crowd flows for all buildings is small at night but skewed at the center of campus in the daytime. Comparing Fig. 15 with Fig. 2, we can also see that the buildings with higher flows have larger attention weights (thus more important).



(a) Spatial attention, 11am – 12pm. (b) Spatial attention, 0am – 1am. (c) Temporal attention at a residential hall.

Fig. 15. Illustration of (a, b) spatial attention head weights for two different time intervals; and (c) 24-hour temporal attention weights (1st attention head) for a target time interval (0–1am on 2021-03-27).

We select the first temporal attention head for building-level outflow embeddings and illustrate its temporal attention weights, *i.e.,* $\boldsymbol{\alpha'}_{k,1}^{(\mathrm{Tp})}[b]$ (Eq. (15) in Sec. 3.4), in Fig. 15c. Specifically, we show the attention weight upon each of the 24 hours ($k \in \{1, \ldots, 24\}$) before the target time interval (0am–1am on 2021-03-27) to be predicted. We note that the $\boldsymbol{\alpha'}_{k,1}^{(\mathrm{Tp})}[b]$ represents the importance of the building-level outflow embeddings of time interval k in all 24 historical time intervals, and the higher value represents the higher correlation between k and the target time interval. We can see higher attention weights in Fig. 15c during two levels of temporal closeness, *i.e.,* approximately 6 hours and 24 hours, which are mainly due to the periodic travel routines and campus activities.

(b) *Transition and Flow Predictions*: We further visualize two cases with predictions and ground-truths in terms of building-cluster transitions and AP-level flows in Figs. 16 and 17. The demonstrated accuracy on the dynamic transition/flow data validates the effectiveness of our model designs.

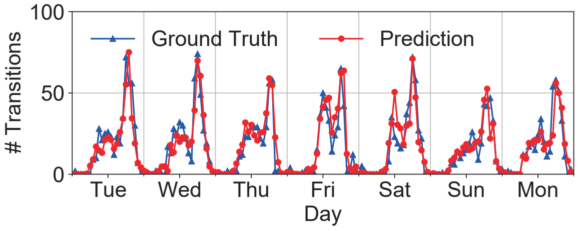


Fig. 16. One-week transitions from a dining hall and a cluster.

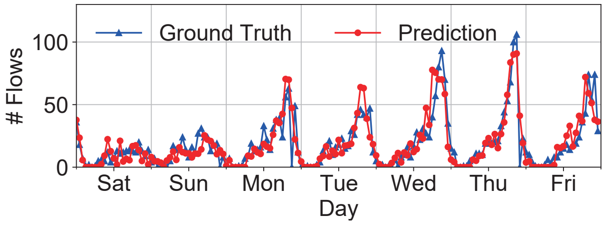


Fig. 17. One-week outflows of an AP in a residential hall.

6 RELATED WORK

We review the related work in the following three categories.

(a) *Flow/Transition Prediction*: Predicting *crowd flows* has become an essential task for many urban applications [16–18]. A pioneering study by Zhang *et al.* [48] predicted the inflows/outflows of different city regions by spatio-temporal residual networks. Lin *et al.* further developed a context-aware spatio-temporal neural network for crowd flow prediction in metropolis [25]. Pan *et al.* proposed a deep meta learning approach to predict the urban traffic flows [31]. Jiang *et al.* achieved prediction of crowd dynamics at big events by an online system which forecasted the trend in the short future only from the current observations [21]. Origin-destination transition prediction has been recently considered for travel time estimation [37] and travel demand prediction [29]. Despite the accuracy achieved, the prior studies [10, 19, 42] cannot fully differentiate the diverse impacts of the spatio-temporal sparsity and skewness. Furthermore, few of these existing flow/transition studies have studied jointly leveraging flows and transitions to mitigate the sparsity in the data.

Our GAEFT fills the gap in studying *predictive* multi-task crowd analytics on flows/transitions by using the Wi-Fi association data. To further mitigate the sparsity and skewness issues within transitions, we propose a novel building clustering technique based on buildings' geo-locations and transition patterns. Different from [23], we design a spatio-temporal clustering algorithm based on affinity propagation to group the buildings into clusters, and accurately predict the crowd transitions between buildings and clusters.

(b) *Graph Neural Networks*: Conventional graph embedding methods for graph neural network often characterize correlations between locations based on spatial closeness [44]. However, existing methods [22, 41, 49] often focus on the graph embeddings *locally* captured by correlations between node features of neighboring nodes or diffusion process towards surrounding nodes, while the *global* impacts on the entire graph network have not been thoroughly considered, making it hard to capture the mobility patterns on the spacious urban sites like the university campuses. While the multi-graph approaches [8, 26] have been studied for different mobility applications, they cannot effectively reflect the hierarchical structures (APs/buildings/clusters) introduced by our spatio-temporal clustering. Instead, we propose a novel spatio-temporal graph attention embedding mechanism to capture and differentiate the spatial and temporal correlations in both local and global scopes, thus enhancing the learnability on the complex crowd mobility data.

(c) *Wi-Fi for Mobility Analysis*: With ubiquitous adoptions of mobile devices, Wi-Fi network data has been studied for various smart mobility applications [12–15]. Sapiezynski *et al.* utilized Wi-Fi data to infer person-to-person proximity [32]. Using Wi-Fi probe data, Traumuel *et al.* modeled the urban movement by network analysis at various street segments [38]. Zhou *et al.* analyzed the crowd behaviors in a social event through data mining of Wi-Fi sensing data [50]. Instead of tracing individuals [39] that is privacy-intrusive, we focus on leveraging Wi-Fi association and dissociation data to *predict* the spatio-temporal transitions and flows of crowds, and our case study insights and findings will help enable proactive and accurate crowd mobility analytics and management.

7 DEPLOYMENT DISCUSSIONS

We discuss the deployment of GAEFT in the following three aspects.

(a) *Privacy*: Via our data preprocessing, all sensitive user-related information has been removed, and both associations/disassociations and transitions have been aggregated before further data analytics. In this work, we aim at predicting the collective and aggregate number of transitions and flows instead of analyzing the mobility patterns of individuals, which preserves the user privacy. Our project has been vetted by the university internal review board (IRB) and no IRB approval is needed due to no privacy concern. Further privacy-preservation enhancement is orthogonal to our studies here and can be referred to [5, 7].

(b) *Impacts of COVID-19 Pandemic*: We note that despite the pandemic impacts on our campus during 2020–2021 academic year, the students, faculty, and staff still had access to the academic buildings. Our core model designs in this study is general, and can be applied upon the crowd flows and transitions collected after the pandemic. Further in-depth studies on the variations of crowd mobility patterns from the Wi-Fi data collected before, during, and after the COVID-19 pandemic will be considered in our future work.

(c) *Incorporation of Other Factors*: In this work, we use the weather conditions and weekends/holidays as the external factors. Other factors such as various campus events held in the buildings/classrooms may influence the transitions and flows [3]. We will further include these factors in our future work.

8 CONCLUSIONS

We propose GAEFT, a novel multi-task graph attention neural network for predicting the crowd transitions and flows based on collective campus Wi-Fi association data. Through crowd analysis, we have identified the sparsity and skewness in the crowd mobility data in addition to the complex spatio-temporal characteristics of the crowd transitions and flows. To address this, we design a novel geographic clustering to group the buildings into neighborhoods with similar transition features, and a novel attention-based graph neural network which captures the spatial and temporal correlations between buildings. GAEFT jointly predicts the transitions between buildings and clusters, and the flows of APs in the buildings. Extensive experimental studies have corroborated the accuracy of GAEFT in predicting the dynamic, complex, and sparse transitions and flows.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their insightful comments. We also thank the University of Connecticut Information Technology Services (UITS) for their assistance in collecting Wi-Fi association data. This project was supported in part by the University of Connecticut Research Excellence Program (REP), NVIDIA Applied Research Accelerator Program, and National Science Foundation under Grant 2118102. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

[1] Serdar Çolak, Antonio Lima, and Marta C González. 2016. Understanding congested travel in urban areas. *Nature Communications* 7, 1 (2016), 1–8.

[2] Allan M De Souza, Roberto S Yokoyama, Guilherme Maia, Antonio Loureiro, and Leandro Villas. 2016. Real-time path planning to prevent traffic jam through an intelligent transportation system. In *Proc. IEEE ISCC*. IEEE, 726–731.

[3] Zipei Fan, Xuan Song, Tianqi Xia, Renhe Jiang, Ryosuke Shibasaki, and Ritsu Sakuramachi. 2018. Online deep ensemble learning for predicting citywide human mobility. *Proc. ACM IMWUT* 2, 3 (2018), 1–21.

[4] Zhihan Fang, Yu Yang, Shuai Wang, Boyang Fu, Zixing Song, Fan Zhang, and Desheng Zhang. 2019. MAC: Measuring the impacts of anomalies on travel time of multiple transportation systems. *Proc. ACM IMWUT* 3, 2 (2019), 1–24.

[5] Jie Feng, Can Rong, Funing Sun, Diansheng Guo, and Yong Li. 2020. PMF: A privacy-preserving human mobility prediction framework via federated learning. *Proc. ACM IMWUT* 4, 1 (2020), 1–21.

[6] Brendan J Frey and Delbert Dueck. 2007. Clustering by passing messages between data points. *Science* 315, 5814 (2007), 972–976.

[7] Chen Gao, Chao Huang, Yue Yu, Huandong Wang, Yong Li, and Depeng Jin. 2019. Privacy-preserving cross-domain location recommendation. *Proc. ACM IMWUT* 3, 1 (2019), 1–21.

[8] Xu Geng, Yaguang Li, Leye Wang, Lingyu Zhang, Qiang Yang, Jieping Ye, and Yan Liu. 2019. Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting. In *Proc. AAAI*, Vol. 33. 3656–3663.

[9] Craig Gentry and Zulfikar Ramzan. 2005. Single-database private information retrieval with constant communication rate. In *International Colloquium on Automata, Languages, and Programming*. Springer, 803–815.

[10] Yongshun Gong, Zhibin Li, Jian Zhang, Wei Liu, and Yu Zheng. 2020. Online spatio-temporal crowd flow distribution prediction for complex metro system. *IEEE TKDE* (2020).

[11] Anhong Guo, Anuraag Jain, Shomiron Ghose, Gierad Laput, Chris Harrison, and Jeffrey P Bigham. 2018. Crowd-AI camera sensing in the real world. *Proc. ACM IMWUT* 2, 3 (2018), 1–20.

[12] Suining He and S.-H. Gary Chan. 2015. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys & Tutorials* 18, 1 (2015), 466–490.

[13] Suining He, Tianyang Hu, and S.-H. Gary Chan. 2015. Contour-based trilateration for indoor fingerprinting localization. In *Proc. ACM SenSys*. 225–238.

[14] Suining He and Kang G. Shin. 2018. Steering Crowdsourced Signal Map Construction via Bayesian Compressive Sensing. In *Proc. IEEE INFOCOM*. 1016–1024.

[15] Suining He and Kang G. Shin. 2019. Crowd-flow graph construction and identification with spatio-temporal signal feature fusion. In *Proc. IEEE INFOCOM*. 757–765.

[16] Suining He and Kang G. Shin. 2019. Spatio-Temporal Capsule-Based Reinforcement Learning for Mobility-on-Demand Network Coordination. In *Proc. WWW* (San Francisco, CA, USA). 2806–2813.

[17] Suining He and Kang G. Shin. 2020. Dynamic Flow Distribution Prediction for Urban Dockless E-Scooter Sharing Reconfiguration. In *Proc. WWW*. 133–143.

[18] Suining He and Kang G. Shin. 2020. Towards fine-grained flow forecasting: A graph attention approach for bike sharing systems. In *Proc. WWW*. 88–98.

[19] Jilin Hu, Bin Yang, Chenjuan Guo, Christian S Jensen, and Hui Xiong. 2020. Stochastic origin-destination matrix forecasting using dual-stage graph convolutional, recurrent neural networks. In *Proc. IEEE ICDE*. 1417–1428.

[20] Tiziano Inzerilli, Anna Maria Vegni, Alessandro Neri, and Roberto Cusani. 2008. A location-based vertical handover algorithm for limitation of the ping-pong effect. In *Proc. IEEE WiMOB*. IEEE, 385–389.

[21] Renhe Jiang, Xuan Song, Dou Huang, Xiaoya Song, Tianqi Xia, Zekun Cai, Zhaonan Wang, Kyoung-Sook Kim, and Ryosuke Shibasaki. 2019. DeepUrbanEvent: A system for predicting citywide crowd dynamics at big events. In *Proc. ACM SIGKDD*. 2114–2122.

[22] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2017. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926* (2017).

[23] Yexin Li, Yu Zheng, Huichu Zhang, and Lei Chen. 2015. Traffic prediction in a bike-sharing system. In *Proc. ACM SIGSPATIAL*. 1–10.

[24] Lei Lin, Zhengbing He, and Srinivas Peeta. 2018. Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach. *Transportation Research Part C: Emerging Technologies* 97 (2018), 258–276.

[25] Ziqian Lin, Jie Feng, Ziyang Lu, Yong Li, and Depeng Jin. 2019. DeepSTN+: Context-aware spatial-temporal neural network for crowd flow prediction in metropolis. In *Proc. AAAI*, Vol. 33. 1020–1027.

[26] Lingbo Liu, Jingwen Chen, Hefeng Wu, Jiajie Zhen, Guanbin Li, and Liang Lin. 2020. Physical-virtual collaboration modeling for intra-and inter-station metro ridership prediction. *IEEE TITS* (2020).

[27] Lingbo Liu, Zhilin Qiu, Guanbin Li, Qing Wang, Wanli Ouyang, and Liang Lin. 2019. Contextualized spatial–temporal network for taxi origin-destination demand prediction. *IEEE TITS* 20, 10 (2019), 3875–3887.

[28] Qian Liu, Dexuan Sha, Wei Liu, Paul Houser, Luyao Zhang, Ruizhi Hou, Hai Lan, Colin Flynn, Mingyue Lu, Tao Hu, et al. 2020. Spatiotemporal patterns of COVID-19 impact on human activities and environment in mainland China using nighttime light and air quality data. *Remote Sensing* 12, 10 (2020), 1576.

[29] Zhicheng Liu, Fabio Miranda, Weiting Xiong, Junyan Yang, Qiao Wang, and Claudio Silva. 2020. Learning geo-contextual embeddings for commuting flow prediction. In *Proc. AAAI*, Vol. 34. 808–816.

[30] Jamie Lopez Bernal, Nick Andrews, Charlotte Gower, Eileen Gallagher, Ruth Simmons, Simon Thelwall, Julia Stowe, Elise Tessier, Natalie Groves, Gavin Dabrera, et al. 2021. Effectiveness of COVID-19 vaccines against the B. 1.617. 2 (Delta) variant. *New England Journal of Medicine* (2021).

[31] Zheyi Pan, Yuxuan Liang, Weifeng Wang, Yong Yu, Yu Zheng, and Junbo Zhang. 2019. Urban traffic prediction from spatio-temporal data using deep meta learning. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1720–1730.

[32] Piotr Sapiezynski, Arkadiusz Stopczynski, David Kofoed Wind, Jure Leskovec, and Sune Lehmann. 2017. Inferring person-to-person proximity using WiFi signals. *Proc. ACM IMWUT* 1, 2 (2017), 1–20.

[33] Jiaxing Shen, Jiannong Cao, and Xuefeng Liu. 2019. BaG: Behavior-aware Group Detection in Crowded Urban Spaces using WiFi Probes. In *Proc. WWW*. 1669–1678.

[34] Shun-Yao Shih, Fan-Keng Sun, and Hung-yi Lee. 2019. Temporal pattern attention for multivariate time series forecasting. *Machine Learning* 108, 8 (2019), 1421–1441.

[35] Yiwei Song, Yunhuai Liu, Wenqing Qiu, Zhou Qin, Chang Tan, Can Yang, and Desheng Zhang. 2020. MIFF: Human Mobility Extractions with Cellular Signaling Data under Spatio-temporal Uncertainty. *Proc. ACM IMWUT* 4, 4 (2020), 1–19.

[36] Waldo R Tobler. 1970. A computer movie simulating urban growth in the Detroit region. *Economic Geography* 46, sup1 (1970), 234–240.

[37] Florian Toqué, Etienne Côme, Mohamed Khalil El Mahrsi, and Latifa Oukhellou. 2016. Forecasting dynamic public transport origin-destination matrices with long-short term memory recurrent neural networks. In *Proc. IEEE ITSC*. 1071–1076.

[38] Martin W Traunmueller, Nicholas Johnson, Awais Malik, and Constantine E Kontokosta. 2018. Digital footprints: Using WiFi probe and locational data to analyze human mobility trajectories in cities. *Computers, Environment and Urban Systems* 72 (2018), 4–12.

[39] Amee Trivedi, Camellia Zakaria, Rajesh Balan, Ann Becker, George Corey, and Prashant Shenoy. 2021. WiFiTrace: Network-based Contact Tracing for Infectious Diseases Using Passive WiFi Sensing. *Proc. ACM IMWUT* 5, 1 (2021), 1–26.

[40] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. NeurIPS*. 5998–6008.

[41] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[42] Yuandong Wang, Hongzhi Yin, Hongxu Chen, Tianyu Wo, Jie Xu, and Kai Zheng. 2019. Origin-destination matrix prediction via graph convolution: a new perspective of passenger demand modeling. In *Proc. ACM SIGKDD*. 1227–1235.

[43] Xi Xiong, Kaan Ozbay, Li Jin, and Chen Feng. 2020. Dynamic origin–destination matrix prediction with line graph neural networks and Kalman filter. *Transportation Research Record* 2674, 8 (2020), 491–503.

[44] Xi Yang, Suining He, and Huiqun Huang. 2020. Station Correlation Attention Learning for Data-driven Bike Sharing System Usage Prediction. In *Proc. IEEE MASS*. 640–648.

[45] Zijun Yao, Yanjie Fu, Bin Liu, Yanchi Liu, and Hui Xiong. 2016. POI recommendation: A temporal matching between POI popularity and user regularity. In *Proc. IEEE ICDM*. 549–558.

[46] Nicholas Jing Yuan, Yu Zheng, Xing Xie, Yingzi Wang, Kai Zheng, and Hui Xiong. 2014. Discovering urban functional zones using latent activity trajectories. *IEEE TKDE* 27, 3 (2014), 712–725.

[47] Huichu Zhang, Yu Zheng, and Yong Yu. 2018. Detecting urban anomalies using multiple spatio-temporal data sources. *Proc. ACM IMWUT* 2, 1 (2018), 1–18.

[48] Junbo Zhang, Yu Zheng, and Dekang Qi. 2017. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proc. AAAI*, Vol. 31.

[49] Chuanpan Zheng, Xiaoliang Fan, Cheng Wang, and Jianzhong Qi. 2020. GMAN: A graph multi-attention network for traffic prediction. In *Proc. AAAI*, Vol. 34. 1234–1241.

[50] Yuren Zhou, Billy Pik Lik Lau, Zann Koh, Chau Yuen, and Benny Kai Kiat Ng. 2020. Understanding crowd behaviors in a social event by passive WiFi sensing and data mining. *IEEE IoT-J* 7, 5 (2020), 4442–4454.