IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 23, 2020, accepted January 4, 2021, date of publication January 11, 2021, date of current version January 19, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3050607

Predicting Emergency Medical Service Demand With Bipartite Graph Convolutional Networks

RUIDONG JIN[1,2], TIANQI XIA[2,3], XIN LIU[2,4], TSUYOSHI MURATA[1,4], AND KYOUNG-SOOK KIM[2,4]

[1]Department of Computer Science, Tokyo Institute of Technology, Tokyo 152-8552, Japan
[2]AIRC, National Institute of Advanced Industrial Science and Technology, Tokyo 135-0064, Japan
[3]Center for Spatial Information Science, The University of Tokyo, Chiba 277-8568, Japan
[4]AIST-Tokyo Tech Real World Big-Data Computation Open Innovation Laboratory, Tokyo 152-8550, Japan

Corresponding author: Xin Liu (xin.liu@aist.go.jp)

This work was supported in part by the JSPS Grant-in-Aid for Early-Career Scientists under Grant 19K20352, in part by the JSPS Grant-in-Aid for Scientific Research under Grant 17H01785, in part by the JST CREST under Grant JPMJCR1687, and in part by the New Energy and Industrial Technology Development Organization.

ABSTRACT Emergency medical service (EMS) plays an essential role in increasing survival rates as it provides first aid to victims of life-threatening emergencies. However, unbalanced EMS supply-demand distribution in the metropolis may cause a shortage of accessible EMS resources and delay the first aid treatment. There is an urgent need to discover the hidden EMS supply-demand relation, predict the incoming EMS demand, and take precautions against unexpected emergencies. This study assumes that EMS demand correlates with population demographic data, regional socioeconomic factors, and hospital conditions. To model these correlated factors, we represent Tokyo's ambulance record data as a hospital-region bipartite graph and propose a bipartite graph convolutional neural network model to predict the EMS demand between hospital-region pairs. Our approach achieves $77.3\% - 87.7\%$ accuracy in binary demand label prediction task. It significantly outperforms traditional machine learning algorithms, statistical models, and the latest graph-based methods. Finally, we use a case study to show the significance of EMS demand forecasting, proving that our approach can contribute to public health emergency management by making EMS predictions.

INDEX TERMS Bipartite graph embedding, binary label classification, demand modeling, emergency medical service, emergency event prediction, graph convolutional network.

I. INTRODUCTION

Emergency Medical Services (EMS) are responsible for providing out-of-hospital medical care to illness and injury patients and transporting them to a medical facility. However, the EMS service is not evenly distributed. For example, the Tokyo Metropolis consists of 23 wards, each with different structures, functions, and population composition. Some regions have high emergency demand but relatively few EMS resources. The unbalanced EMS resource may raise the risk of a shortage of available EMS resources and result in the delay of first aid treatment, especially when some high-priority medical emergencies such as sudden cardiac arrest or a big event as the Tokyo Olympic Games takes place. Besides, EMS is time-sensitive, i.e., the later the service arrives at the incident sites, the more severe the damage to the patient's health will be. Thus, accurate prediction of EMS demand can help provide quick and efficient medical treatment and increase the survival rate for elderly patients [1].

Moreover, Japan is experiencing a ''super-aging'' society. People aged 65 and older who currently make up a quarter of the total population are estimated to reach a third by 2050. The seniors tend to have a high EMS demand for their declining immune function and physical function. Thus, Japan urgently needs to discover the hidden relationship of EMS supply-demand, predict the incoming EMS demand, and take precautions against unexpected emergencies. Therefore, we are motivated to focus on Tokyo, the largest metropolis in ''super-aging'' society Japan, to study the relationship between EMS supply and demand. Predicting the incoming EMS demand in advance would induce a better allocation of

The associate editor coordinating the review of this manuscript and approving it for publication was Pasquale De Meo.

resources and be of considerable significance to public health emergency management.

Previous research on EMS demand prediction can be classified into model-based and data-driven approaches. The model-based approaches predict specific demand by applying the total capacity with multiple rules such as supply-demand ratio and distance decay [2]. Model-based approaches work well in an actual situation and easy to understand. However, these approaches rely on predefined rules and cannot clearly explain the anomalies [3]. On the other hand, the data-driven approaches take the demand prediction problem as a classification or regression problem based on the observational data and employ statistical methods or naive machine learning algorithms for the solution. Data-driven approaches can fit complicated situations where the mathematical model is difficult to establish. Specifically, some statistical methods utilize social demographic data, socioeconomic factors, and land use factors to improve EMS performance in a particular city [1], [4], [5]. Moreover, predict EMS demand over time by a spatio-temporal statistical model raises recent attention. The spatio-temporal statistical model utilizes location information and time-series information to predict time and location accurately [6]–[8]. However, in complex real-world situations, the statistical model may be complicated with too high order or too much non-linearity, or even unavailable, where huge factors need to be considered. It should be noted that previous researches overwhelmingly focus on EMS demand prediction in specific regions, while few works study the demand between regions and hospitals. Hospitals play a vital role in the medical emergency. Accurate EMS demand prediction between regions and hospitals is meaningful to urban public health emergency management and better EMS resource allocation.

With the development of artificial intelligence, graph embedding methods have been proved a better performance than the traditional methods with flat inputs [9], [10], which indicates a new way of predicting EMS demand via the demand-supply relation graph. And this topic in bipartite graphs is seldom studied in substantial research on learning graph data [11]–[16]. Those graph embedding methods learn graph nodes as low-dimensional vector representations. With the help of node representation vectors, subsequent graph problems like node classification, link prediction, and node clustering can be easily solved by combining them with some existing machine learning methods. Start with DeepWalk [17] and Node2vec [18], many graph embedding architectures are developed. Specifically, graph convolutional networks (GCN) [19]–[21] has been proved an efficient neural network architecture regarding graph embedding researches. Many methods re-define the convolution operation for graph structure data and are developed as members of GCN based models. The core concept of GCN is iteratively aggregating feature information from a node's neighborhood nodes in a graph. Compared with other graph embedding methods that focus on graph contents, GCN utilizes both node features and holds the potential of exploiting the graph topology structure. Intuitively, a hypothesis is suggested that the thinking of GCN may also perform well in modeling the EMS demand-supply relation graph and give an accurate prediction regarding the EMS demand between hospitals and regions.

With the concerns mentioned above, in this article, we study the EMS demand-supply relation between regions and hospitals, then propose an approach for predicting the EMS demand at the hospital-region level. Our motivation is based on the idea that the EMS demand between a region and a hospital is mainly affected by population demographics, regional socioeconomic factors, and hospital conditions. Thus, we collect data from various sources, including regional demographic data (daytime population number / census population number / crime number), regional land-use data (industrial / residential / commercial area ratio), regional historical emergency data (illness / injury case number), hospital information data (number of beds and doctors / past number of patients), and transportation data (distance between region and hospital).

We develop a Bipartite Graph Convolutional Network (BiGCN) model that exploits the multi-modal features of the data for EMS demand prediction. Specifically, we transform the demand prediction problem to an edge label classification problem in a hospital-region bipartite graph, as shown in Fig. 1. The bipartite graph is a particular type of graph whose nodes divide into two disjoint sets such that the edge connects nodes from one set to the other. Hospitals and regions serve as two individual node sets. Hospitals and regions serve as two individual node sets. The edge connects a hospital node and a region node, indicating that an emergency happened in this region, and the injured people are sent to this hospital. Feature attributes are attached to each node and each edge.

Notably, we find that traditional GCN does not work correctly in bipartite graphs because it confuses the information from the two disjoint node sets. Our BiGCN model separates the convolution operation of the two node sets and overcomes the shortcomings of traditional GCN. The experimental results demonstrate that our approach achieves $77.3\% - 87.7\%$ accuracy in the label prediction task, which is significantly superior to baseline traditional machine learning algorithms, statistical models, and the latest graph-based methods. Finally, We use a case study to illustrate that our approach can provide valuable suggestions for allocating injured people in emergencies. It proves that our work is meaningful to urban public health emergency management, make the public aware of the significance of EMS demand prediction, and help local governments better allocate EMS resource and decrease the emergency risk.

The main contributions of this article are summarized as follows:

- We are the first to analyze EMS demand at the hospital-region level in a metropolis like Tokyo. Our work is of considerable significance to public health emergency management.

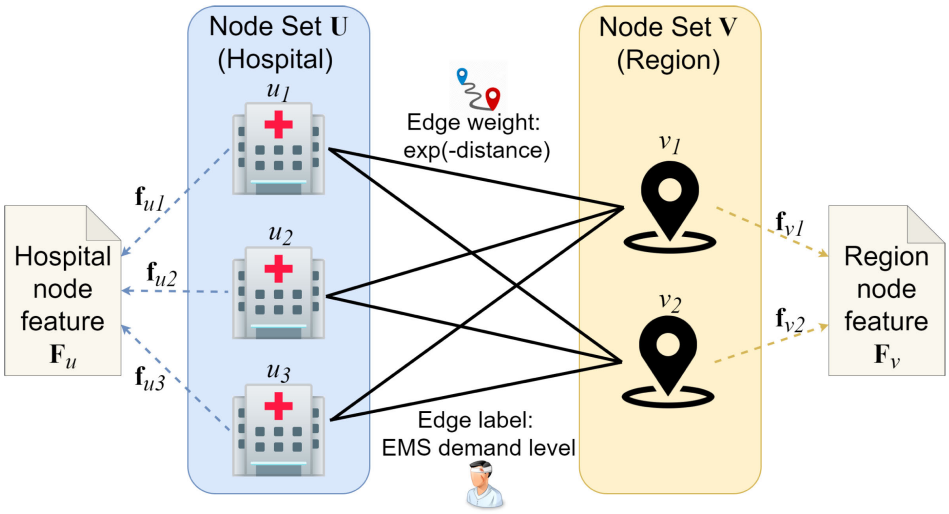


FIGURE 1. The hospital-region EMS bipartite graph.

- We represent the ambulance record data as a hospital-region bipartite graph and transform the EMS demand prediction problem to an edge label classification problem in the bipartite graph.
- We analyze the limitations of GCN in bipartite graphs and propose BiGCN by fully considering the structure characteristics of bipartite graphs. BiGCN is not limited to the hospital-region bipartite graph in this article but has the potential to become a general model for learning node embeddings and accomplishing supervised learning tasks in non-specific bipartite graphs.
- We conduct experiments and demonstrate that our approach achieves excellent performance in the demand label prediction task and significantly outperforms baseline methods, including traditional machine learning algorithms, statistical models, and state-of-the-art graph-based methods. We discover the main factors that affect the EMS demand prediction most. We use a case study to show how our approach can contribute to public health emergency management.

SOURCE CODE

The Python implementation of the BiGCN model is provided for reviewing: https://github.com/Tracy-King/BiGCN.

The rest of this article is organized as follows. Section II summarizes the recent literature on EMS demand analysis and graph learning. Section III introduces the problem statement of EMS demand prediction. Section IV discusses the limitations of the state-of-the-art GCN in the bipartite graph. Section V presents our BiGCN model. Section VI reports experiment results and presents a case study. Finally, Section VII concludes our research.

II. RELATED WORK

We present the related work from two different angles: EMS demand analysis and graph embedding.

A. EMS DEMAND ANALYSIS

Past research on EMS supply and demand analysis can be classified into model-based and data-driven approaches. The model-based approaches predict specific demand by applying the total capacity with multiple rules such as supply-demand ratio and distance decay [2]. The main topic is accessibility as an important indicator in evaluating the justice of medical service [22]. Moreover, the large variance of accessibility indicates an unequal spatial distribution of EMS facilities. Therefore, EMS facility location optimization (FLO) is another popular topic. There are several types of EMS facilities for FLO models, including emergency devices [23], emergency centers [24], and the ambulance stations [25].

The data-driven approaches take the demand prediction problem as a classification or regression problem based on the observational data and employ a statistical or naive machine learning model for the solution. Specifically, the statistics-based approaches perform well in spatial EMS analysis [26]. uEMS [4] has the object to maximize the EMS vehicle coverage with limited ambulance stations in an urban area and provides a generalized linear model to locate the urban EMS ambulance station. Steins *et al.* [27] use a Zero-Inflated Poisson(ZIP) regression approach to develop a statistical EMS demand forecasting model. Grekousis and Liu [5] propose a spatial-based Artificial Neural Networks(ANN) approach that identifies the geographical location of expected emergency events. Besides, predicting EMS demand over time has raised recent attentions [7], [28], [29]. Some spatio-temporal approaches are proposed to utilize both location information and time-series information since EMS prediction needs to be accurate for both time and location. Setzler *et al.* [6] design an ANN to forecast EMS demand volume of specific areas during different times of the day. Channouf *et al.* [8] develop and compare several regression models to analyze the time-series information of a major Canadian city's daily and hourly EMS

call volume. Zhou *et al.* [30] introduce a Gaussian Mixture Model(GMM) to estimate the ambulance demand distribution in Toronto, Canada. However, previous research overwhelmingly focuses on EMS demand prediction in specific regions, while few studies study the demand between regions and hospitals. Hospitals play a vital role in the medical emergency. Accurate EMS demand prediction between regions and hospitals is meaningful to urban public health emergency management and better EMS resource allocation.

B. GRAPH EMBEDDING

Graphs are used in many science branches to represent the patterns of connections between the components of complex systems. There is a surge of interest in graph embedding [11]–[16]. The goal is to learn a mapping that embeds nodes as points in a low-dimensional vector space. The learned embeddings can be taken as feature inputs for downstream machine learning tasks, and this technology has achieved great success in many applications. Researchers have proposed various graph embedding methods such as matrix factorization [31], edge reconstruction [32], random walks plus skip-gram model [17], [18], and graph neural networks [19], [33], [34].

However, graph embedding in bipartite graphs is relatively less studied. The bipartite graph is a ubiquitous data structure to model the relationship between two types of entities. Examples of bipartite graphs include the author-paper graph, the customer-product graph, the player-event graph, the actor-movie graph, and the keyword-document graph. As far as we know, there are few works on bipartite graph embedding. BiNE is an approach by performing biased random walks and preserving the long-tail distribution of nodes in bipartite graphs [35]. BGNN is another approach that utilizes inter-domain message passing and intra-domain alignment towards information fusion [36]. However, these two methods do not precisely solve our problem for two reasons. First, BiNE only employs the graph structure information but not the node feature information. Second, both are unsupervised learning methods that do not fully utilize the training set to optimize the models, resulting in suboptimal performance.

On the other hand, there are extensive researches on graph embedding for heterogeneous graphs, which contain multi-typed nodes or multi-typed edges. Note that bipartite graphs that contain two typed nodes can be regarded as a particular class of heterogeneous graphs. Metapath2vec [37] uses meta-path based random walks to aggregate the heterogeneous neighborhood nodes and then exploit a skip-gram model to learn the node embeddings. HAN [38] is a heterogeneous graph neural network based on the hierarchical attention mechanism, including node-level and semantic-level attentions. It fully considers the importance of node neighbors and different meta-paths. HetGNN [39] combines heterogeneous structural information and node attributes. It has an excellent performance in graphs with multiple types of nodes and edges.

TABLE 1. Symbols and interpretations.

Symbol	Interpretation
$\mathbf{U}, \mathbf{V}$	Hospital and region node set
$\mathbf{E}$	Edge set
u_i, v_j	Node in $\mathbf{U}$ and $\mathbf{V}$
M, N	Number of nodes in $\mathbf{U}$ and $\mathbf{V}$
e_{ij}	Edge between u_i and v_j
K	Number of edges $\lvert\mathbf{E}\rvert$
$\mathbf{F}_u, \mathbf{F}_v$	Feature matrix for $\mathbf{U}$ and $\mathbf{V}$
$\mathbf{f}_{u_i}, \mathbf{f}_{v_j}$	Feature vector for u_i and v_j
P, Q	Number of features for $\mathbf{U}$ and $\mathbf{V}$
y_{ij}	Label of the edge between node u_i and v_j
$\mathbf{Y}$	Edge label set
$\mathbf{B}_u, \mathbf{B}_v$	Weighted incidence matrix for $\mathbf{U}$ and $\mathbf{V}$
$\mathbf{D}_u, \mathbf{D}_v$	Diagonal degree matrix for $\mathbf{U}$ and $\mathbf{V}$
$\mathbf{H}_u^{(t)}, \mathbf{H}_v^{(t)}$	Node embedding matrix for $\mathbf{U}$ and $\mathbf{V}$ at the tth layer
$\mathbf{h}_{u_i}^{(t)}, \mathbf{h}_{v_j}^{(t)}$	Node embedding vector for u_i and v_j at the tth layer
$\mathbf{H}_e^{(t)}$	Edge embedding matrix at the tth layer
$\mathbf{h}_{e(ij)}^{(t)}$	Edge embedding vector for e_{ij} at the tth layer
t, T	Layer number in models
c	Dimension scaling parameter

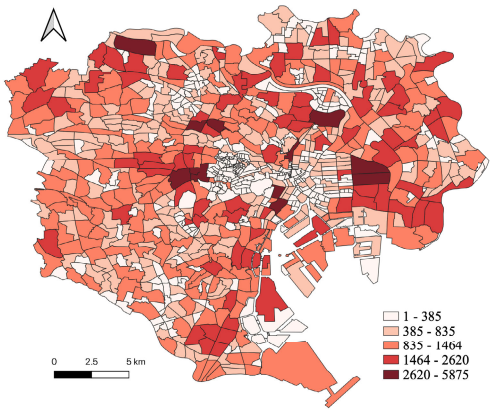


FIGURE 2. The number of emergency cases in different regions of Tokyo.

III. PROBLEM SETUP

This section first describes the dataset we used in our research. It then represents the EMS data as a hospital-region bipartite graph and transforms the problem to an edge label classification problem in the graph. Table 1 lists the main notations in this article.

A. DATASET DESCRIPTION

This study collects a dataset of ambulance records from 1st January 2017 to 31st December 2017 for the Central Tokyo area. The dataset is provided by the Tokyo Fire Department. It contains 624,062 emergency cases. Each record includes the patient's age, gender, ambulance types (disease or injury), ambulance scene, and hospital address.

There are 931 administrative regions and 291 hospitals that can provide emergency medical care in Tokyo. Administrative regions are divided by following the official district division principle of government. The map of regions with the number of cases is visualized in Fig. 2. We are interested in the emergency demand from one region to one hospital.

TABLE 2. Features of hospitals and regions.

Node Type	Feature Category	Feature Name
Region	Population	Day time population Census population Crime number
	Land-Use Zoning	Industrial area ratio Residential area ratio Commercial area ratio
	Historical Emergency Number (for patients sent to hospitals)	Total case number Injury case number Disease case number
	Historical Emergency Number (for patients not sent to hospitals)	Total case number Injury case number Disease case number
Hospital	Capacity	Number of beds Number of doctors
	Historical Emergency Number	Total case number Injury case number Disease case number

The raw data aggregates into 270,921 hospital-region pairs, with only 23,308 pairs having demand. We attach a binary label denoting the low/high demand level for each pair. The label is determined by the percentage of the demand in the capacity of the respective hospital. As a result, 13,603 pairs are labeled "high", and 9,705 pairs are labeled "low". Moreover, we collect a bunch of feature information for the regions and hospitals, as listed in Table 2. Also, we follow [2] to measure the Euclidean distance between regions and hospitals.

B. EDGE LABEL CLASSIFICATION IN THE HOSPITAL-REGION GRAPH

We represent the EMS data as a bipartite graph $\mathbf{G} = (\mathbf{U}, \mathbf{V}, \mathbf{E})$, as shown in Fig. 1. $\mathbf{U}$ and $\mathbf{V}$ are the node sets for hospitals and regions, respectively. $M = |\mathbf{U}|$ and $N = |\mathbf{V}|$ are the number of nodes in the two sets. $\mathbf{E} \subseteq \mathbf{U} \times \mathbf{V}$ denotes the edge set. $K = |\mathbf{E}|$ is the number of edges. $e_{ij} \in \mathbf{E}$ represents the EMS demand relationship between the hospital node $u_i \in \mathbf{U}$ and region node $v_j \in \mathbf{V}$. The edge weight is

$$\omega_{ij} = \exp(-\text{distance}(u_i, v_j)), \tag{1}$$

which indicates the "closeness" of u_i and v_j. *distance* $(\cdot)$ denotes the Euclidean distance between two node, and $\exp(\cdot)$ denotes the exponential function.

The hospital node u_i and region node v_j are associated with feature vectors $\mathbf{f}_{u_i} \in \mathcal{R}^P$ and $\mathbf{f}_{v_j} \in \mathcal{R}^Q$, which are based on preprocessed raw features in Table 2, and P and Q are the respective number of features. $\mathbf{F}_u = [\mathbf{f}_{u_1}, \ldots, \mathbf{f}_{u_M}]^\top \in \mathcal{R}^{M\times P}$ and $\mathbf{F}_v = [\mathbf{f}_{v_1}, \ldots, \mathbf{f}_{v_N}]^\top \in \mathcal{R}^{N\times Q}$ denote the feature matrix of hospitals and regions. Moreover, $y_{ij} \in \{-1, 1\}$ is the label for edge e_{ij} (which corresponds to low/high demand between hospital and region) and $\mathbf{Y} = \{y_{ij} | e_{ij} \in \mathbf{E}\}$ is the set of edge labels.

Since the edge connects nodes between $\mathbf{U}$ and $\mathbf{V}$, the adjacency matrix $\mathbf{A}$ takes a block off-diagonal form

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}_{M\times M} & \mathbf{B}_u \\ \mathbf{B}_v & \mathbf{0}_{N\times N} \end{bmatrix}, \tag{2}$$

TABLE 3. Statistics of the bipartite graph for the EMS dataset.

Item	Statistics
Number of Hospital Nodes M	291
Number of Region Nodes N	931
Number of Edges K	23,308
Number of Hospital Features P	5
Number of Region Features Q	12
Average/Maximal Degree of Hospital Nodes	80.1/616
Average/Maximal Degree of Region Nodes	25.0/92

where $\mathbf{B}_u \in \mathcal{R}^{M\times N}$ and $\mathbf{B}_v \in \mathcal{R}^{N\times M}$ are the incidence matrix for $\mathbf{U}$ and $\mathbf{V}$, respectively. $\mathbf{B}_u = \mathbf{B}_v^\top$, and $\mathbf{B}_{u(i,j)} = \mathbf{B}_{v(j,i)} = \omega_{ij}$. $\mathbf{D}$ is the degree diagonal matrix:

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_u & \mathbf{0}_{M\times N} \\ \mathbf{0}_{N\times M} & \mathbf{D}_v, \end{bmatrix}, \tag{3}$$

where $\mathbf{D}_u = \text{Diag}(\sum_i \mathbf{B}_{u(1,i)}, \ldots, \sum_i \mathbf{B}_{u(M,i)})$ and $\mathbf{D}_v = \text{Diag}(\sum_i \mathbf{B}_{v(1,i)}, \ldots, \sum_i \mathbf{B}_{v(N,i)})$ are the diagonal degree matrices of $\mathbf{U}$ and $\mathbf{V}$.

Table 3 reports the statistics of the hospital-region bipartite graph. The problem in Section III can be represented as an edge label classification problem in the graph. Suppose $\mathbf{Y}$ is split into a training set $\mathbf{Y}^{\text{train}}$, a validation set $\mathbf{Y}^{\text{val}}$, and a test set $\mathbf{Y}^{\text{test}}$. Then, our problem (as illustrated in Fig. 3) is set up as follows: given the features of regions and hospitals($\mathbf{F}_u$, $\mathbf{F}_v$), the distance of hospital-region pairs($\mathbf{A}$), and a portion ($10\% - 80\%$) of known labels($\mathbf{Y}^{\text{train}}$, and $\mathbf{Y}^{\text{val}}$), how can we predict the remaining edge labels($\mathbf{Y}^{\text{test}}$).

IV. LIMITATIONS OF GCN IN BIPARTITE GRAPHS

In this section, we discuss the limitations of traditional GCN and explain why it cannot correctly work in the bipartite graph.

GCN [19] is a type of neural network architectures that can leverage the graph structure and node features for graph analysis, and this model has achieved great success in many tasks. GCN consists of aggregators and updaters. The aggregator gathers information guided by the graph

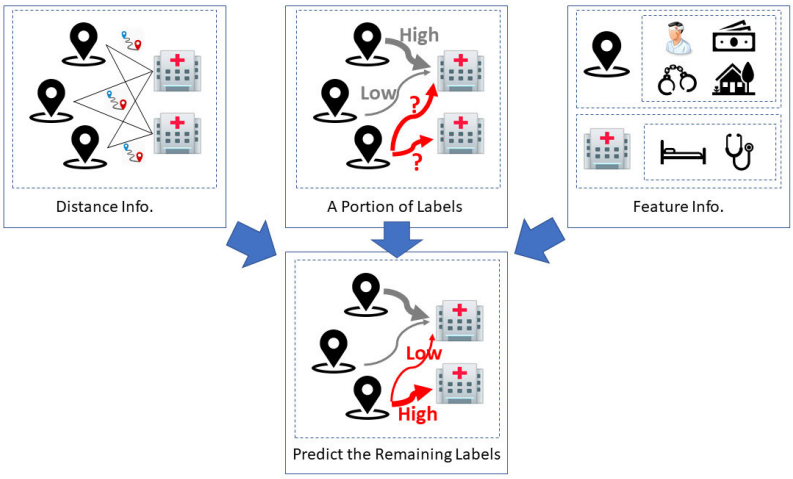


FIGURE 3. Predicting the high/low demand label of hospital-region pairs.

structure, and the updater updates nodes' hidden states according to the gathered information. The core of the method is learning node representations, or *embeddings*, in a low-dimensional vector space that encode information about the graph. Specifically, the convolutional layer is based on the following equation:

$$\mathbf{H}^{(t+1)} = \sigma(\mathbf{L}\mathbf{H}^{(t)}\mathbf{W}^{(t+1)}) \tag{4}$$

where $\mathbf{L}$ is the aggregation matrix, $\mathbf{H}^{(t)}$ is the node embedding matrix in t-th layer, $\mathbf{H}^{(0)}$ is initialized with the node feature matrix, $\mathbf{W}^{(l)}$ is the trainable weight matrix in l-th layer, and $\sigma(\cdot)$ is the activation function.

GCN is originally designed for unipartite graphs, and it adopts the symmetric normalized Laplacian matrix $\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$ as the aggregator $\mathbf{L}^{\text{GCN}}$, where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ and $\tilde{\mathbf{D}}$ is the degree diagonal matrix of $\tilde{\mathbf{A}}$. Here $\mathbf{I}$ denotes the identity matrix. In the following, we analyze the limitations of applying GCN in bipartite graphs. To simplify the analysis, we consider an approximate form $\mathbf{L}^{\text{GCN}} \approx \mathbf{D}^{-\frac{1}{2}}\tilde{\mathbf{A}}\mathbf{D}^{-\frac{1}{2}}$. Moreover, we have to concatenate the two feature matrices $\mathbf{F}_u$ and $\mathbf{F}_v$ as the initialization of the node embedding matrix $\mathbf{H}^{(0)}$. However, since the two matrices have different column dimensions and $P < Q$, we have to expand $\mathbf{F}_u$ with zero values to make an alignment:

$$\mathbf{H}^{(0)} = \begin{bmatrix} \mathbf{F}_u & | & \mathbf{0}_{M\times(Q-P)} \\ & & \mathbf{F}_v \end{bmatrix} \tag{5}$$

where | is an auxiliary symbol for matrix partitions, and $\mathbf{0}$ is a zero matrix with the given shape. Then, the first convolutional layer in Eq. (4) becomes

$$\mathbf{H}^{(1)} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{A}+\mathbf{I})\mathbf{D}^{-\frac{1}{2}}\mathbf{H}^{(0)}\mathbf{W}^{(1)} \tag{6}$$

$$= \begin{bmatrix} \left(\mathbf{D}_u^{-\frac{1}{2}}\mathbf{B}_u\mathbf{D}_v^{-\frac{1}{2}}\mathbf{F}_v\right) + \left(\mathbf{D}_u^{-1}\mathbf{F}_u \mid \mathbf{0}_{M\times(Q-P)}\right) \\ \left(\mathbf{D}_v^{-\frac{1}{2}}\mathbf{B}_v\mathbf{D}_u^{-\frac{1}{2}}\mathbf{F}_u \mid \mathbf{0}_{N\times(Q-P)}\right) + \left(\mathbf{D}_v^{-1}\mathbf{F}_v\right) \end{bmatrix} \mathbf{W}^{(1)} \tag{7}$$

From Eq. (7), we can find that the aggregator sums the hospital feature matrix $\mathbf{F}_u$ and the region feature matrix $\mathbf{F}_v$. Note that $\mathbf{F}_u$ and $\mathbf{F}_v$ refer to information from different sources, which implies that they have different dimensions and they are based on different scaling systems. Therefore, it is not reasonable to sum them together. For the same reason, it is insufficient to use the same matrix $\mathbf{W}^{(1)}$ for the filter operation. As a result, GCN does not work correctly in bipartite graphs. A new approach is in demand to wisely apply the thinking of GCN to bipartite graphs.

V. PROPOSED APPROACH

In this section, we propose an approach to address the problem in the last section. We hypothesize that the EMS demand between a region and a hospital can be predicted by modeling the demographic information and the distance of regions and hospitals. To overcome the disadvantage of traditional GCN and apply the thinking of GCN in bipartite graphs, we introduce an improved GCN based model, bipartite graph convolutional network structure BiGCN. Fig. 4 illustrates the overview structure of BiGCN model. The core method of BiGCN is learning the embedding representation of nodes by graph convolution operation, then learning the edge embedding representations based on its two side node embeddings, finally predicting the edge labels based on the edge embeddings. The spotlight of BiGCN is the utilization of both the graph structure information and node features, and the individual operation for two disjoint node sets. The model evaluation depends on the accuracy and f1 score metrics regarding the EMS demand prediction task.

A. BiGCN

Based on the structural characteristics of bipartite graphs that $\mathbf{U}$ and $\mathbf{V}$ are disjoint node sets with distinct properties (degree distributions, associated features), our idea is to separate the convolution operation for $\mathbf{U}$ and $\mathbf{V}$. Specifically, the

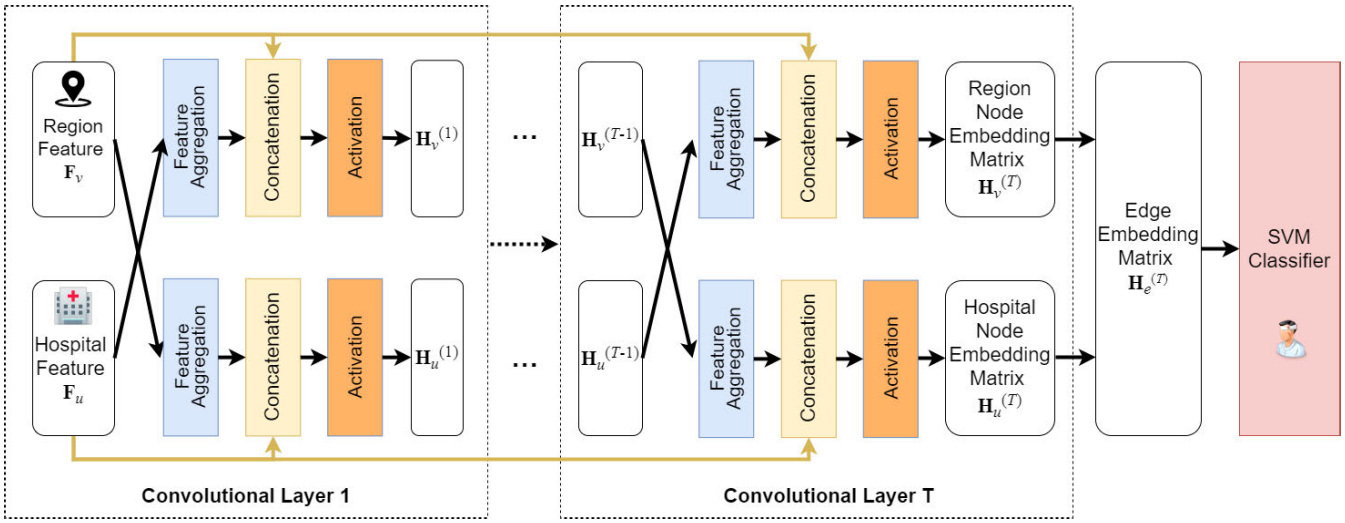


FIGURE 4. Overview of the structure of our approach.

mathematical expression of our convolutional layer is

$$\mathbf{H}_u^{(0)} = \mathbf{F}_u \tag{8}$$

$$\mathbf{H}_v^{(0)} = \mathbf{F}_v \tag{9}$$

$$\mathbf{H}_u^{(t+1)} = \sigma\left(\left[\mathbf{D}_u^{-1}\mathbf{B}_u\mathbf{H}_v^{(t)}\mathbf{W}_u^{(t+1)} \parallel \mathbf{F}_u\boldsymbol{\omega}_u^{(t+1)}\right]\right) \tag{10}$$

$$\mathbf{H}_v^{(t+1)} = \sigma\left(\left[\mathbf{D}_v^{-1}\mathbf{B}_v\mathbf{H}_u^{(t)}\mathbf{W}_v^{(t+1)} \parallel \mathbf{F}_v\boldsymbol{\omega}_v^{(t+1)}\right]\right) \tag{11}$$

$\mathbf{H}_u^{(t)}$ and $\mathbf{H}_v^{(t)}$ are the learned node embedding matrix in the t-th layer for $\mathbf{U}$ and $\mathbf{V}$, respectively. $\mathbf{W}_u^{(t)}$, $\mathbf{W}_v^{(t)}$, $\boldsymbol{\omega}_u^{(t)}$, and $\boldsymbol{\omega}_v^{(t)}$ are trainable weight matrices (filters) in t-th layer. $\parallel$ is the concatenation operation.

The process of the convolutional layer contains three steps, as shown in Fig. 4. Take node $u_i \in \mathbf{U}$ as an example. We first aggregate features from its neighbors in $\mathbf{V}$ and update the aggregated features by a filter (this step corresponds to $\mathbf{D}_u^{-1}\mathbf{B}_u\mathbf{H}_v^{(t)}\mathbf{W}_u^{(t+1)}$ in Eq.(10)). In parallel, we update the original features of u_i by another filter (this step corresponds to $\mathbf{F}_u\boldsymbol{\omega}_u^{(t+1)}$ in Eq.(10)). Finally, we concatenate the updated features from two sources and pass them to an activation function. Similar process applies for node $v_i \in \mathbf{V}$.

There are three differences between GCN and BiGCN. First, GCN only use one aggregator and one filter for $\mathbf{U}$ and $\mathbf{V}$, while BiGCN enforces independent aggregators ($\mathbf{D}_u^{-1}\mathbf{B}_u$, $\mathbf{D}_v^{-1}\mathbf{B}_v$) and filters ($\mathbf{W}_u^{(t)}$, $\mathbf{W}_v^{(t)}$, $\boldsymbol{\omega}_u^{(t)}$ for the two node sets. Secondly, GCN intentionally adding self-connections ($\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$) to preserve the feature of the node itself. On the other hand, BiGCN uses the concatenation operation for this purpose and overcomes the shortcomings of simply summing the features from different sources. Thirdly, GCN uses the symmetric normalized Laplacian matrix $\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$ as the aggregator. However, the node degree distributions of $\mathbf{U}$ and $\mathbf{V}$ are compositionally distinct from each other, and thus the symmetric normalization is not meaningful. Instead, BiGCN uses the random walk Laplacian matrix $\mathbf{D}_u^{-1}\mathbf{B}_u$ and $\mathbf{D}_v^{-1}\mathbf{B}_v$ to take the average of neighboring node features in the aggregation process. All of the above imply that BiGCN fully considers the structure characteristics of bipartite graphs.

We explain about the dimensionality. Note that in the hidden layer, we concatenate the features from $\mathbf{U}$ and $\mathbf{V}$. We hope to keep the dimension in proportion to the original dimension of the two sources. Therefore, we assume $\lceil c^{(t)}P\rceil + \lceil c^{(t)}Q\rceil$ is the dimension of t-th hidden units. $c^{(t)} \in \mathcal{R}^+$ is a dimension scaling parameter. $\lceil\cdot\rceil$ is the ceiling function. Following this assumption, we have:

$$\mathbf{H}_u^{(t)} \in \mathcal{R}^{M\times(\lceil c^{(t)}P\rceil+\lceil c^{(t)}Q\rceil)} \tag{12}$$

$$\mathbf{H}_v^{(t)} \in \mathcal{R}^{N\times(\lceil c^{(t)}P\rceil+\lceil c^{(t)}Q\rceil)} \tag{13}$$

$$\mathbf{W}_u^{(1)} \in \mathcal{R}^{Q\times\lceil c^{(1)}Q\rceil} \tag{14}$$

$$\mathbf{W}_v^{(1)} \in \mathcal{R}^{P\times\lceil c^{(1)}P\rceil} \tag{15}$$

$$\mathbf{W}_u^{(t)} \in \mathcal{R}^{(\lceil c^{(t-1)}P\rceil+\lceil c^{(t-1)}Q\rceil)\times\lceil c^{(t)}Q\rceil}, \quad \text{for } t \geq 2 \tag{16}$$

$$\mathbf{W}_v^{(t)} \in \mathcal{R}^{(\lceil c^{(t-1)}P\rceil+\lceil c^{(t-1)}Q\rceil)\times\lceil c^{(t)}P\rceil}, \quad \text{for } t \geq 2 \tag{17}$$

$$\boldsymbol{\omega}_u^{(t)} \in \mathcal{R}^{P\times\lceil c^{(t)}P\rceil} \tag{18}$$

$$\boldsymbol{\omega}_v^{(t)} \in \mathcal{R}^{Q\times\lceil c^{(t)}Q\rceil} \tag{19}$$

Next, we analyze the computational complexity of the convolutional layer for GCN and BiGCN. With regard to Eqs.(7), (10), (11), the complexity is dominated by matrix multiplication. We equally assume that both models have the same dimension D for the hidden units. Moreover, we suppose $P < Q$. In the case of a sparse bipartite graph, the complexity of GCN is $\mathcal{O}(KQ+KP+(M+N)QD) = \mathcal{O}(KQ+(M+N)QD)$, while the complexity of BiGCN is $\mathcal{O}(KQ + MQD + MPD + KP + NPD + NQD) = \mathcal{O}(KQ + (M + N)QD)$. In the case of a dense bipartite graph, the complexity of GCN is $\mathcal{O}(MNQ + NMP + (M + N)QD) = \mathcal{O}(MNQ + (M + N)QD)$, while the complexity of BiGCN is $\mathcal{O}(MNQ + MQD + MPD + NMP + NPD + NQD) = \mathcal{O}(MNQ + (M + N)QD)$. Therefore, GCN and BiGCN have the same computational complexity.

B. LOSS FUNCTION

We now turn to edge label classification. We have obtained node embedding matrix $\mathbf{H}_u^{(T)} = [\mathbf{h}_{u_1}^{(T)}, \ldots, \mathbf{h}_{u_M}^{(T)}]^\top$ and $\mathbf{H}_v^{(T)} = [\mathbf{h}_{v_1}^{(T)}, \ldots, \mathbf{h}_{v_N}^{(T)}]^\top$, where T is the number of layers in BiGCN, and $\mathbf{h}_{u_i}^{(T)}$ and $\mathbf{h}_{v_j}^{(T)}$ are the embedding for u_i and v_j. For an edge e_{ij}, we can generate its embedding $\mathbf{h}_{e(ij)}^{(T)} \in \mathcal{R}^{(\lceil c^{(T)}P\rceil+\lceil c^{(T)}Q\rceil)}$ using the Hadamard product [18]

$$[\mathbf{h}_{e(ij)}^{(T)}]_l = [\mathbf{h}_{u_i}^{(T)}]_l \cdot [\mathbf{h}_{v_j}^{(T)}]_l, \tag{20}$$

where $l \in 1, \cdots, (\lceil c^{(T)}P\rceil + \lceil c^{(T)}Q\rceil)$ denotes the subscript of the l-th element. In this way, we can generate an edge embedding matrix $\mathbf{H}_e^{(T)} = [\mathbf{h}_{e_1}^{(T)}, \ldots, \mathbf{h}_{e_K}^{(T)}]^\top \in \mathcal{R}^{K\times(\lceil c^{(T)}P\rceil+\lceil c^{(T)}Q\rceil)}$.

Then, we use Support Vector Machine (SVM) classifier to predict the binary edge labels. Specifically, the predicted labels can be formulated as

$$\hat{\mathbf{y}} = \operatorname{sgn}\left(\mathbf{H}_e^{(T)}\boldsymbol{\omega}\right), \tag{21}$$

where $\boldsymbol{\omega} \in \mathcal{R}^{(\lceil c^{(T)}P\rceil+\lceil c^{(T)}Q\rceil)}$ is a trainable vector and sgn is the sign function.

For model training, we evaluate the classification error over all examples in the training set based on the hinge loss function:

$$\mathcal{L}_c = \mathcal{H}(\mathbf{H}_e^{(T)}) \tag{22}$$

$$= \frac{1}{|\mathbf{Y}^{\text{train}}|} \sum_{k\in\mathcal{I}(\mathbf{Y}^{\text{train}})} \max\left(0, 1 - \mathbf{y}_k\left(\mathbf{H}_e^{(T)}\boldsymbol{\omega}\right)_k\right) + \frac{1}{2}\|\boldsymbol{\omega}\|_2^2, \tag{23}$$

where $\mathbf{y}$ is the vector of edge labels and $\mathcal{I}(\mathbf{Y}^{\text{train}})$ is the set of edge indices in $\mathbf{Y}^{\text{train}}$. Moreover, to alleviate the information loss in the hidden layer of BiGCN, we formulate a recurrent loss by applying the hinge loss to the Hadamard product of hidden units

$$\mathcal{L}_r = \sum_{t=1}^{T-1} \mathcal{H}(\mathbf{H}_e^{(t)}) \tag{24}$$

Finally, the total loss is

$$\mathcal{L} = \mathcal{L}_c + \alpha\mathcal{L}_r, \tag{25}$$

where α is a hyperparameter for balancing $\mathcal{L}_c$ and $\mathcal{L}_r$.

VI. EXPERIMENT

We conducted experiments on the edge label classification task to answer the following questions regarding BiGCN model:

- Does BiGCN agree with our hypothesis and perform well in bipartite graphs? How does it compare with other models?
- Which input feature influences the prediction result most?
- How many layers should be used in BiGCN?
- Is recurrent loss necessary? How does it impact model performance?
- How can our approach be used for public health emergency management?

In the following, we first explain experimental settings and baselines. After that, we discuss the results.

EXPERIMENT SETUP

We randomly split $\mathbf{Y}$ into $\mathbf{Y}^{\text{train}}$, $\mathbf{Y}^{\text{val}}$, and $\mathbf{Y}^{\text{test}}$. $\mathbf{Y}^{\text{val}}$ accounts for 10%, $\mathbf{Y}^{\text{test}}$ ratio ranges from 10% $\sim$ 80%, and the remaining is for $\mathbf{Y}^{\text{train}}$. As for the implementation of BiGCN, we chose Parametric Rectified Linear Unit (PReLU) as the activation function. We applied batch normalization with a momentum of 0.9 to each convolutional layer. We initialized the weight parameters using He initialization [40]. We trained the model by Adam optimizer [41] with a learning rate of 0.01. We set the maximum training iteration as 500 and applied an early stopping strategy if the validation loss does not decrease for 10 iterations. The implementation is in Python and PyTorch on Google Colaboratory platform with Intel(R) Xeon(R) CPU @ 2.20GHz, NVIDIA Tesla P100 16GB GPU, and 25GB memory.

BASELINES

We consider traditional machine learning classifiers, statistical models, and the latest graph-based methods such as graph embedding and graph neural networks as baselines. The details are listed below.

1) **SVM** (Support Vector Machine), **GBDT** (Gradient Boosting Decision Tree), and **LR** (Logistic Regression): These are traditional machine learning classifiers.
2) **GMM** [30]: This is a Gaussian Mixture Model approach.
3) **ZIP** [27]: This is a Zero-Inflated Poisson regression approach.
4) **Node2Vec** [18]: This is an unsupervised algorithm for learning graph node embeddings based on random walks and skip-gram model.
5) **GCN** [19]: This is one of the most widely used graph convolutional networks. This method, as stated in Section IV, can be interpreted as smoothing the node features in the neighborhoods guided by the graph structure.
6) **VGAE** [42]: This is a variational graph autoencoder, in which GCN is used as an encoder to learn node embeddings.
7) **GraphSAGE** [33]: This is another popular method for generating node embeddings based on graph structure and node features.
8) **BiNE** [35]: This is an extension of unsupervised graph embedding algorithm to bipartite graphs.
9) **BGNN** [36]: This is a bipartite graph neural network for learning node embeddings in an unsupervised fashion.
10) **Metapath2vec** [37]: This is an extension of conventional graph embedding technique to heterogeneous graphs, which contain multi-typed nodes or multi-typed edges. This approach formalizes

TABLE 4. The results for predicting the hospital-region labels by different approaches.

Model	Test Ratio									
	10%		20%		40%		60%		80%	
	Acc.	F1.	Acc.	F1.	Acc.	F1.	Acc.	F1.	Acc.	F1.
SVM	0.734	0.769	0.716	0.759	0.715	0.750	0.707	0.747	0.696	0.737
GBDT	0.812	0.843	0.796	0.840	0.815	0.844	0.782	0.832	**0.778**	**0.830**
LR	0.624	0.529	0.628	0.539	0.602	0.489	0.637	0.564	0.582	0.443
GMM	0.667	0.198	0.555	0.407	0.645	0.470	0.632	0.454	0.623	0.385
ZIP	0.851	0.592	0.785	0.706	0.596	0.047	0.592	0.037	0.591	0.051
Node2Vec	0.756	0.791	0.762	0.800	0.758	0.800	0.751	0.794	0.745	0.788
GCN	0.730	0.767	0.718	0.753	0.728	0.764	0.729	0.760	0.700	0.736
VGAE	0.736	0.785	0.723	0.775	0.725	0.776	0.728	0.776	0.694	0.738
GraphSAGE	0.422	0.584	0.411	0.583	0.411	0.582	0.576	0.729	0.585	0.738
BiNE	0.687	0.743	0.686	0.738	0.683	0.737	0.679	0.744	0.678	0.738
BGNN	0.682	0.743	0.687	0.709	0.678	0.744	0.676	0.731	0.676	0.734
Metapath2vec	0.718	0.761	0.729	0.773	0.719	0.765	0.717	0.764	0.711	0.753
HetGNN	0.666	0.745	0.665	0.741	0.656	0.738	0.669	0.742	0.656	0.740
HAN	0.657	0.726	0.679	0.729	0.680	0.734	0.674	0.741	0.678	0.739
BiGCN	**0.877**	**0.895**	**0.860**	**0.881**	**0.848**	**0.871**	**0.827**	**0.852**	0.773	0.804
Performance Gain (%)	3.0 | 107.8	6.1 | 352.0	8.0 | 109.2	4.8 | 116.4	4.0 | 106.3	3.1 | 1853.2	5.7 | 43.5	2.4 | 2302.7	-0.6 | 32.8	-3.1 | 1576.5

meta-path-based random walks to construct the heterogeneous neighborhood of a node to learn embeddings. Note that

11) **HetGNN** [39]: This is a graph neural network model for learning node vector representations in heterogeneous graphs.
12) **HAN** [38]: This is a heterogeneous graph neural network based on the hierarchical node-level and semantic-level attentions.

We employ traditional machine learning classifiers (SVM, GDBT, LR) to predict the hospital-region bipartite graph edge labels with flat inputs consist of distance information and node features. The two baseline statistical models (GMM, ZIP) are designed for EMS demand forecasting in the prior works. The statistical model is much more understandable and simpler to implement than machine learning models. In our experiment, two models receive flat inputs consist of distance information and node features. Graph structure information is not involved. The graph-based approaches (Node2Vec, GCN, VGAE, GraphSAGE, BiNE, BGNN, Metapath2vec, HetGNN, HAN) receive both graph topology structure and node features as input. They are initially designed for only learning node embeddings. Thus, after obtaining the node embeddings by these baseline methods, we generate edge embeddings by applying Hadamard product to these node embeddings and finally feed them to an SVM classifier to predict the hospital-region labels.

A. PERFORMANCE COMPARISON

Table 4 displays the results by BiGCN (with loss balancing hyperparameter $\alpha = 0.1$, dimension of hidden unit $\lceil c^{(t)}P \rceil + \lceil c^{(t)}Q \rceil = 50$, and number of convolution layer $T = 3$) and the baselines. We evaluated the performance in terms of accuracy and F1 score. BiGCN demonstrates the best overall performance. In particular, it achieves the highest accuracy and F1 score when the test ratio ranges from 10% $\sim$ 60%, with 2.4% - 8.0% performance gain over GBDT that is consistently better than the other baselines. The only exception arises when the test ratio reaches 80%, or the training set only accounts for 10%, where our model suffers from an overfitting problem. GBDT, as an additive model, has a good strategy to prevent overfitting and thus obtains superior performance.

It is interesting to note that BiGCN outperforms the other graph-based approaches by a large margin. We attribute this to two reasons. The first reason is the deliberate network structure of BiGCN that considers the characteristics of bipartite graphs. In contrast, GCN, VGAE, and GraphSAGE are designed for unipartite graphs and do not adapt to bipartite graphs. The second reason is the end-to-end model optimization based on our proposed cost function. Conversely, Node2Vec, VGAE, BiNE, and BGNN are unsupervised learning methods to preserve the original graph's topology structure. They thus do not fully utilize the training set to optimize the models. Note that Metapath2vec,

HetGNN, and HAN are originally proposed for heterogeneous graphs, and we find that they also do not fit well with bipartite graphs. For example, although HAN utilizes the node-level and semantic-level attention mechanism, the trivial structure and single meta-path in bipartite graph restrict its performance. As for the statistical models, GMM gives similar performances in all test ratios, while ZIP's performance is severely impacted by the training data volume. It gives an average performance when there is enough training data (test ratio $= 10\% \sim 20\%$), although still lower than BiGCN. However, with the training data decreasing, they tend to give an extraordinarily unbalanced prediction result, i.e., predicting many "high" labels and a few "low" labels. It covers almost all the "high" cases but seldom predicts "low" cases correctly. This result leads to a high precision and a low recall, resulting in a low f1 score in total. This result may be caused by the low adaptability of statistical models in a complicated situation.

B. MAIN FACTORS

The neural network based approach BiGCN is a "black box" and lacks interpretability. On the other hand, although having a lower accuracy, statistical models allow for an understanding of the main factors that influence the prediction. The BiGCN model input involves regional demographic information, land-use information, historical emergency information, hospital information, and distance information. Some information may have a significant impact on EMS demand prediction, and some may have less effect. There is a need for an understanding of the main factors that affect the model output most. The discovery of the main factors in EMS demand prediction contributes to better EMS demand estimation.

Using the statistical method ZIP, we can obtain the coefficient distribution for each variable. The coefficients are interpreted as the ones in a standard Poisson model: the excepted value of dependent variable changes by $\exp(coef.)$ for each unit increase in the corresponding variable. We do a z-test on variable coefficients. Then mainly use two statistical terms, z-score and p-value, to evaluate the variable coefficients. The z-test is a statistical test that indicates whether a variable exhibits statistical significance or exhibits a random pattern. Specifically, in this experiment, z-scores are the standard deviations of variable coefficients, and p-values are probabilities that the variable coefficients are created by some random process. Suppose a variable coefficient has a very high($z > +2.5$) or very low($z < -2.5$) z-score, associated with very small p-values ($p < 0.01$). In that case, this variable is likely to be statistically significant clustering or dispersion, which means it is positively/negatively correlated with the output. The larger the $|z|$ is, the stronger the relationship is.

Table 5 demonstrates the z-scores and p-values of the variable coefficient for each feature. The upper part is the regional feature, and the lower part is the hospital feature. As a result, almost all the features significantly impact the prediction result($|z| > 2.5$ and $p < 0.01$). Specifically, "Euclidean distance"($z = 941.555$), regional "EMS injury case number"($z = -898.339$) and regional "EMS disease case number"($z = 898.346$) are three main factors that affect the EMS demand prediction result most. Features "EMS injury/disease case number sent to hospital" and "EMS injury/disease case number not sent to hospital" are linearly correlated with "EMS injury/disease case number", thus having similar z-scores. Besides, it is noticeable that the hospital "EMS injury case number" feature has $z = 0.250$ and $p = 0.803$. It means that this feature is likely to be randomly distributed and has less influence on the prediction result.

TABLE 5. Z-score and p-value of variable coefficients in ZIP regression model (test ratio=0.1).

Variable name	z-score	p-value
Day time population	-24.406	0.000
Census population	62.036	0.000
EMS total case number	25.681	0.000
EMS injury case number	-898.339	0.000
EMS disease case number	898.346	0.000
Crime Number	22.873	0.000
Residential area ratio	-5.358	0.000
Business area ratio	-5.358	0.000
Industry area ration	-5.358	0.000
Ems total case number sent to hospital	26.367	0.000
EMS injury case number sent to hospital	898.338	0.000
EMS disease case number sent to hospital	-898.347	0.000
EMS total case number not sent to hospital	25.652	0.000
EMS injury case number not sent to hospital	898.339	0.000
EMS disease case number not sent to hospital	-898.346	0.000
Number of doctors	4.145	0.000
Number of beds	-20.783	0.000
EMS total case number	9.806	0.000
EMS injury case number	0.250	0.803
EMS disease case number	-12.376	0.000
EMS total case number in main wards	16.803	0.000
EMS injury case number in main wards	-20.059	0.000
EMS disease case number in main wards	-15.373	0.000
Euclidean distance	941.555	0.000

C. PARAMETER STUDY

1) BALANCING PARAMETER α

As introduced in Section V-B, the loss function is a combination of classification loss $\mathcal{L}_c$ and recurrent loss $\mathcal{L}_r$, with parameter α as a balancing weight. We conducted experiments to investigate how α affects performance.

Fig. 5(a) and 5(b) present the results with α ranging from 0.001 to 1.5. A small α such as 0.001 indicates placing a dominant emphasis on $\mathcal{L}_c$ and ignoring $\mathcal{L}_r$ in the cost, while a high α implies including more weight on $\mathcal{L}_r$ in the cost. We can find that the performance improves as α increases from 0.001 to 0.1, then it gradually declines. Our model reaches a high accuracy at around $\alpha = 0.1$, where recurrent loss $\mathcal{L}_r$ positively contributes to the performance.

2) NUMBER OF CONVOLUTIONAL LAYERS

We studied the influence of the number of convolutional layers on performance. Fig. 5(c) and 5(d) report the results. The best results are obtained with three layers. There is a

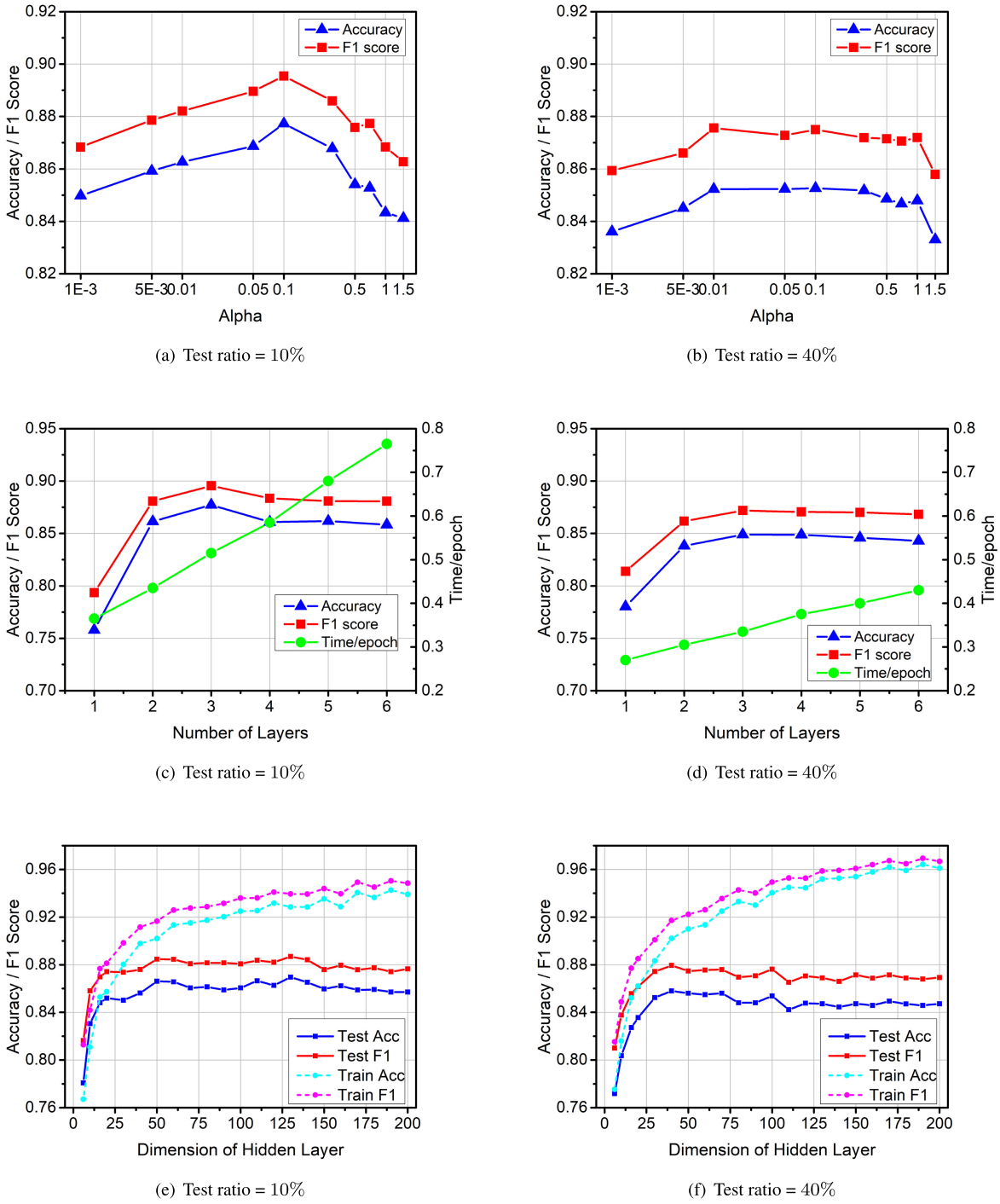


FIGURE 5. **Parameter study. (a)(b) Results with different α. (c)(d) Results with different numbers of layers. (e)(f) Results with different dimensions of hidden units.**

modest decrease in performance when more layers are used because the learned embeddings are over-smoothed [43]. Over-smoothing means if a neural network has many convolutional layers, the output node features may become indistinguishable and give a poor performance. Also, the running time grows as the number of layers increases.

3) DIMENSION OF HIDDEN UNIT

We analyzed the impact of the dimension of hidden units introduced as $\lceil c^{(T)}P \rceil + \lceil c^{(T)}Q \rceil$ in Section V-A. Fig. 5(e) and 5(f) demonstrate the training and testing performance for hidden dimension ranging from 6 to 200. With the increasing hidden dimension, the test performance keeps

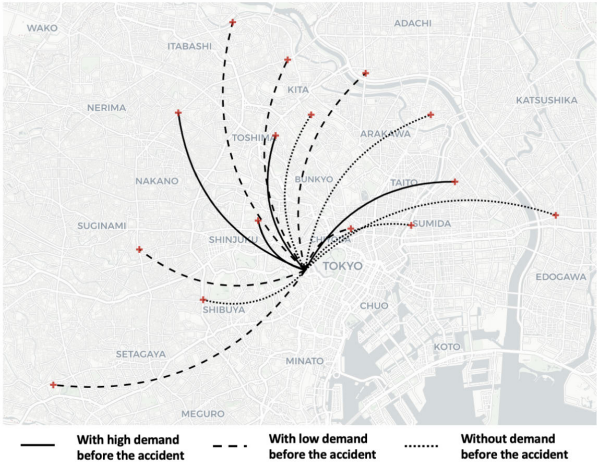


FIGURE 6. The identified hospitals with high EMS demand.

increasing until the dimension reaches around 50, then it becomes relatively stable. However, the training performance still improves even after 50, and gradually goes far beyond the test performance, finally becomes stable after 170. The figure shows that when the hidden dimension exceeds 50, the model begins to be overfitting. The increasing hidden dimension can only fit the training set better instead of improving the test accuracy. When the increasing hidden dimension exceeds around 170, it can no longer improve the training performance. It indicates that the model has achieved its best with the current architecture and cannot easily be improved by merely adding the hidden dimension.

D. CASE STUDY

Finally, we demonstrate an application of our approach in the case of a sudden emergency. We imagine a severe accident occurring in a region named ''Hirakawacho'', which is close to the home of many government agencies such as the National Diet Building and the Prime Minister's Official Residence. The region initially connects to four hospitals with high demand. Then, we modified the region feature *injury number* by adding 1,000 cases to simulate an injury accident where 1,000 people need first aid treatment. We used our model to predict the hospitals with ''high'' demand after the accident according to the new feature input.

The result is shown in Fig. 6. The red cross mark denotes the hospitals, and the center point indicates the region. Our model identifies eleven additional hospitals with high demand, including six hospitals with low demand and five hospitals without demand before the accident. This result is reasonable because we can see that the newly identified hospitals are all close to the region and have high capacity. Such a result is of great value for suggesting the allocation of injured people. It also helps public health emergency management in preparation for similar emergencies in the future.

VII. CONCLUSION

We analyzed the ambulance record data for Tokyo and presented the first approach to predict the EMS demand at the hospital-region level. We represented the data as a hospital-region bipartite graph and proposed a novel BiGCN model to predict the EMS demand between hospital-region pairs. Our approach achieves excellent performance on the prediction accuracy, outperforming the baselines, including traditional machine learning algorithms, statistical models, and the latest graph-based methods by a large margin. We applied a case study to prove the feasibility of the BiGCN model in a real-world situation. Our work is meaningful to urban public health emergency management, make the public aware of the significance of EMS demand prediction, and help local governments better allocate EMS resources and decrease the emergency risk.

ACKNOWLEDGMENT

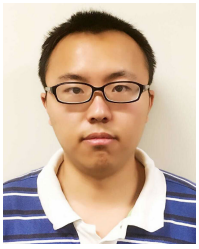
(Ruidong Jin and Tianqi Xia contributed equally to this work.)

REFERENCES

[1] R. Aringhieri, G. Carello, and D. Morale, ''Supporting decision making to improve the performance of an italian emergency medical service,'' *Ann. Oper. Res.*, vol. 236, no. 1, pp. 131–148, Jan. 2016.
[2] J. E. Anderson, ''The gravity model,'' *Annu. Rev. Econ.*, vol. 3, no. 1, pp. 133–160, 2011.
[3] Z.-S. Hou and Z. Wang, ''From model-based control to data-driven control: Survey, classification and perspective,'' *Inf. Sci.*, vol. 235, pp. 3–35, Jun. 2013.
[4] M. Amorim, S. Ferreira, and A. Couto, ''Road safety and the urban emergency medical service (uEMS): Strategy station location,'' *J. Transp. Health*, vol. 6, pp. 60–72, Sep. 2017.
[5] G. Grekousis and Y. Liu, ''Where will the next emergency event occur? Predicting ambulance demand in emergency medical services using artificial intelligence,'' *Comput., Environ. Urban Syst.*, vol. 76, pp. 110–122, Jul. 2019.
[6] H. Setzler, C. Saydam, and S. Park, ''EMS call volume predictions: A comparative study,'' *Comput. Oper. Res.*, vol. 36, no. 6, pp. 1843–1851, Jun. 2009.
[7] Z. Zhou and D. S. Matteson, ''Predicting ambulance demand: A spatio-temporal kernel approach,'' in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2015, pp. 2297–2303.
[8] N. Channouf, P. L'Ecuyer, A. Ingolfsson, and A. N. Avramidis, ''The application of forecasting techniques to modeling emergency medical system calls in Calgary, Alberta,'' *Health Care Manage. Sci.*, vol. 10, no. 1, pp. 25–45, Jan. 2007.
[9] S. A. Fadaee and M. Amir Haeri, ''Classification using link prediction,'' *Neurocomputing*, vol. 359, pp. 395–407, Sep. 2019.
[10] B. Taskar, M.-F. Wong, P. Abbeel, and D. Koller, ''Link prediction in relational data,'' in *Proc. NeurIPS*, 2004, pp. 659–666.
[11] H. Cai, V. W. Zheng, and K. Chen-Chuan Chang, ''A comprehensive survey of graph embedding: Problems, techniques, and applications,'' *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 9, pp. 1616–1637, Sep. 2018.
[12] P. Cui, X. Wang, J. Pei, and W. Zhu, ''A survey on network embedding,'' *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 5, pp. 833–852, May 2019.
[13] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, ''Graph neural networks: A review of methods and applications,'' 2018, *arXiv:1812.08434*. [Online]. Available: http://arxiv.org/abs/1812.08434
[14] Z. Zhang, P. Cui, and W. Zhu, ''Deep learning on graphs: A survey,'' 2018, *arXiv:1812.04202*. [Online]. Available: http://arxiv.org/abs/1812.04202
[15] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, ''A comprehensive survey on graph neural networks,'' 2019, *arXiv:1901.00596*. [Online]. Available: http://arxiv.org/abs/1901.00596
[16] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, ''Graph convolutional networks: A comprehensive review,'' *Comput. Social Netw.*, vol. 6, no. 1, p. 11, Dec. 2019.
[17] B. Perozzi, R. Al-Rfou, and S. Skiena, ''DeepWalk: Online learning of social representations,'' in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 701–710.
[18] A. Grover and J. Leskovec, ''Node2vec: Scalable feature learning for networks,'' in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 855–864.

[19] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*. [Online]. Available: http://arxiv.org/abs/1609.02907
[20] R. van den Berg, T. N. Kipf, and M. Welling, "Graph convolutional matrix completion," 2017, *arXiv:1706.02263*. [Online]. Available: http://arxiv.org/abs/1706.02263
[21] F. Monti, M. Bronstein, and X. Bresson, "Geometric matrix completion with recurrent multi-graph neural networks," in *Proc. NeurIPS*, 2017, pp. 3697–3707.
[22] J. Luo, "Integrating the huff model and floating catchment area methods to analyze spatial access to healthcare services," *Trans. GIS*, vol. 18, no. 3, pp. 436–448, Jun. 2014.
[23] B. Bonnet, D. Gama Dessavre, K. Kraus, and J. E. Ramirez-Marquez, "Optimal placement of public-access AEDs in urban environments," *Comput. Ind. Eng.*, vol. 90, pp. 269–280, Dec. 2015.
[24] F. Silva and D. Serra, "Locating emergency services with different priorities: The priority queuing covering location problem," *J. Oper. Res. Soc.*, vol. 59, no. 9, pp. 1229–1238, Sep. 2008.
[25] Z.-H. Zhang and H. Jiang, "A robust counterpart approach to the bi-objective emergency medical service design problem," *Appl. Math. Model.*, vol. 38, no. 3, pp. 1033–1040, Feb. 2014.
[26] R. Aringhieri, M. E. Bruni, S. Khodaparasti, and J. T. van Essen, "Emergency medical services and beyond: Addressing new challenges through a wide literature review," *Comput. Oper. Res.*, vol. 78, pp. 349–368, Feb. 2017.
[27] K. Steins, N. Matinrad, and T. Granberg, "Forecasting the demand for emergency medical services," in *Proc. 52nd Hawaii Int. Conf. Syst. Sci.*, 2019, pp. 1855–1864.
[28] G. S. Zaric, *Operations Research and Health Care Policy*. New York, NY, USA: Springer, 2013. [Online]. Available: https://link.springer.com/book/10.1007%2F978-1-4614-6507-2#editorsandaffiliations
[29] A. Y. Chen, T.-Y. Lu, M. H.-M. Ma, and W.-Z. Sun, "Demand forecast using data analytics for the preallocation of ambulances," *IEEE J. Biomed. Health Informat.*, vol. 20, no. 4, pp. 1178–1187, Jul. 2016.
[30] Z. Zhou, D. S. Matteson, D. B. Woodard, S. G. Henderson, and A. C. Micheas, "A spatio-temporal point process model for ambulance demand," *J. Amer. Stat. Assoc.*, vol. 110, no. 509, pp. 6–15, Jan. 2015.
[31] M. Ou, P. Cui, J. Pei, Z. Zhang, and W. Zhu, "Asymmetric transitivity preserving graph embedding," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1105–1114.
[32] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: Large-scale information network embedding," in *Proc. 24th Int. Conf. World Wide Web*, May 2015, pp. 1067–1077.
[33] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. NeurIPS*, 2017, pp. 1024–1034.
[34] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*. [Online]. Available: http://arxiv.org/abs/1710.10903
[35] M. Gao, L. Chen, X. He, and A. Zhou, "BiNE: Bipartite network embedding," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, p. 715.
[36] C. He, T. Xie, Y. Rong, W. Huang, J. Huang, X. Ren, and C. Shahabi, "Bipartite graph neural networks for efficient node representation learning," 2019, *arXiv:1906.11994*. [Online]. Available: http://arxiv.org/abs/1906.11994
[37] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 135–144.
[38] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, "Heterogeneous graph attention network," in *Proc. World Wide Web Conf.*, May 2019, pp. 2022–2032.
[39] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, "Heterogeneous graph neural network," in *Proc. KDD*, 2019, pp. 793–803.
[40] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.
[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[42] T. N. Kipf and M. Welling, "Variational graph auto-encoders," 2016, *arXiv:1611.07308*. [Online]. Available: http://arxiv.org/abs/1611.07308
[43] Q. Li, Z. Han, and X.-M. Wu, "Deeper insights into graph convolutional networks for semi-supervised learning," in *Proc. AAAI*, 2018, pp. 3538–3545.

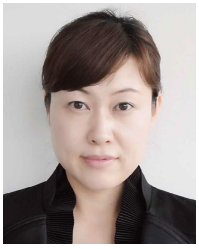
RUIDONG JIN received the B.S. degree in computer science and engineering from Shanghai Jiao Tong University in 2018 and the M.S. degree in artificial intelligence from the Tokyo Institute of Technology in 2020, where he is currently pursuing the Ph.D. degree majoring in artificial intelligence. He is currently a Research Assistant with the Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology. His research interests include graph embedding and graph neural networks.



TIANQI XIA received the Ph.D. degree in environment science from the Graduate School of Frontier Sciences, Center for Spatial Information Science, The University of Tokyo, in 2020. He was a Research Assistant with the Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology. His current research interests include spatiotemporal data mining, applied GIS and big data analysis on public health, human mobility analysis, and inbound tourism analysis.

XIN LIU received the master's degree in computer science from Wuhan University and the Ph.D. degree in computer science from the Tokyo Institute of Technology. He is currently a Senior Researcher with the Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology. His main research interests are graph and network analytics, data mining, machine learning, and deep learning.

TSUYOSHI MURATA is currently a Professor with the Department of Computer Science, School of Computing, Tokyo Institute of Technology. He has been involved in research on artificial intelligence, especially network science, machine learning, and Web mining.



KYOUNG-SOOK KIM received the B.S., M.S., and Ph.D. degrees in computer science from Pusan National University, South Korea, in 1998, 2001, and 2007, respectively. She is currently a Senior Research with the Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology, Japan. She currently serves the Co-Chair of Moving Features SWG and GeoAI DWG of the Open Geospatial Consortium and an Expert of ISO TC204/WG3 and JTC1 SC42/WG2. Her research interests are in big data analysis and spatiotemporal data platforms.