sustainability

MDPI

Article

Multi-Modal Graph Interaction for Multi-Graph Convolution Network in Urban Spatiotemporal Forecasting

Lingyu Zhang [1,2,3], Xu Geng [4], Zhiwei Qin [2], Hongjun Wang [5], Xiao Wang [2,3], Ying Zhang [2,3], Jian Liang [2,3], Guobin Wu [2,3], Xuan Song [5] and Yunhai Wang [1,*]

1 School of Computer Science and Technology, Shandong University, Qingdao 250012, China
2 Didi Chuxing, Beijing 065001, China
3 Data Science and Artificial Intelligence Department, Draweast Tech, Beijing 065001, China
4 Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hongkong 999077, China
5 Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China
* Correspondence: cloudseawang@gmail.com

Abstract: Graph convolution network-based approaches have been recently used to model region-wise relationships in region-level prediction problems in urban computing. Each relationship represents a kind of spatial dependency, such as region-wise distance or functional similarity. To incorporate multiple relationships into a spatial feature extraction, we define the problem as a multi-modal machine learning problem on multi-graph convolution networks. Leveraging the advantage of multi-modal machine learning, we propose to develop modality interaction mechanisms for this problem in order to reduce the generalization error by reinforcing the learning of multi-modal coordinated representations. In this work, we propose two interaction techniques for handling features in lower layers and higher layers, respectively. In lower layers, we propose grouped GCN to combine the graph connectivity from different modalities for a more complete spatial feature extraction. In higher layers, we adapt multi-linear relationship networks to GCN by exploring the dimension transformation and freezing part of the covariance structure. The adapted approach, called multi-linear relationship GCN, learns more generalized features to overcome the train–test divergence induced by time shifting. We evaluated our model on a ride-hailing demand forecasting problem using two real-world datasets. The proposed technique outperforms state-of-the art baselines in terms of prediction accuracy, training efficiency, interpretability and model robustness.

Keywords: multi-modal machine learning; graph convolution networks; multi-task learning; transfer learning

Citation: Zhang, L.; Geng, X.; Qin, Z.; Wang, H.; Wang, X.; Zhang, Y.; Liang, J.; Wu, G.; Song, X.; Wang, Y. Multi-Modal Graph Interaction for Multi-Graph Convolution Network in Urban Spatiotemporal Forecasting. *Sustainability* **2022**, *14*, 12397. https://doi.org/10.3390/su141912397

Academic Editor: Hong Tang

Received: 2 August 2022
Accepted: 20 September 2022
Published: 29 September 2022

1. Introduction

The deployment of urban sensor networks is one of the most important progresses in the urban digitization process. Recent advances in sensor technology enable the collection of a large variety of datasets. Multi-modality is one of the most significant features in the knowledge discovery process in urban computing. Data from different sources are often correlated with each other. For region-level prediction problems, such as crowd flow prediction [1,2] or taxi demand prediction [3–5], it has become a common practice to incorporate a large variety of auxiliary datasets, such as weather, Point of Interests (also known as POI) , road network and events. In this paper, we define each auxiliary dataset as a modality and study multi-modal learning on multi-graph convolution networks (MGCN) for spatiotemporal prediction problems in urban computing. This task is challenging due to complex spatial dependencies and a temporal shifting generalization gap.

Designing a spatial feature extraction method is challenging due to complex region-wise spatial dependencies. GCN-based models [6,7] are first used for traffic prediction on

road networks. Geng et al. [5] proposed Multi-GCN (MGCN) for generic spatiotemporal prediction tasks by stacking three GCNs. Each GCN encodes a unique modality (relationship) of auxiliary data (geo-distance, POI similarity and road network) as graph and extract spatial dependencies from such relationship.

The spatial feature extraction by MGCN architecture is incomplete due to the lack of cross-graph connectivities. Figure 1 shows an example for MGCN. Consider the vertex (region) pair A and D. According to graph topology, A and D are disconnected in all three graphs. MGCN is incapable of extracting features from D for A or vice versa. However, we argue that the A-D relationship is important. The region pair A_3-B_3 and B_2-D_2 are closely related on road connectivity and POI similarity. A and D are related region pairs for spatial feature extraction. To complete the physical meaning for spatial feature extraction by MGCN, the ideal graph connectivity is shown in Figure 1b. It is produced by merging all edges from separate graphs, so that any random walk path is a compound of any kind of relationships.

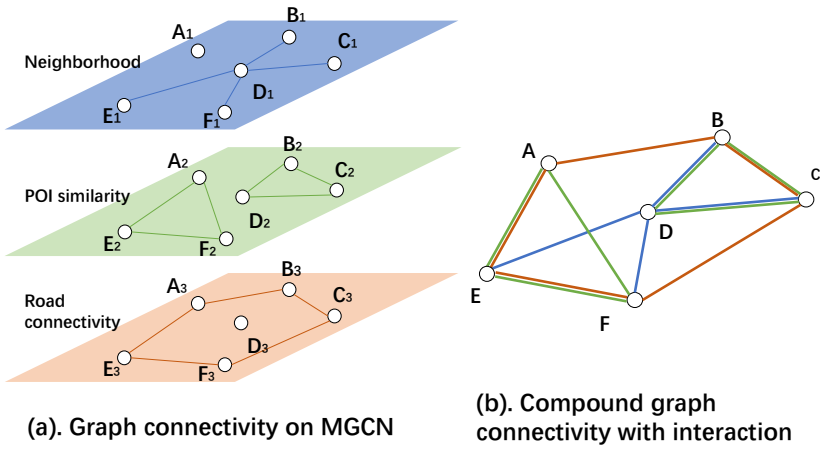

Figure 1. (**a**) shows graph connectivity for MGCN [5] in each graph. X_i represents the vertex (region) X on the i-th graph. Weighted edges between vertices denote a region-wise relationship. There is no interaction among graphs. (**b**) shows compound graph connectivity by adding a graph-wise interaction to MGCN. Vertices are connected as long as there exists an edge in any graph.

Improving model generality to overcome the temporal shifting generalization gap is another challenging task. The temporal pattern for time series data varies along with time. Formally,

$$P(X_t|X_{t-1}, X_{t-2}, \ldots) \neq P(X_{t'}|X_{t'-1}, X_{t'-2}, \ldots), t \neq t'$$

The gap above defines the divergence between temporal pattern distributions in two different time windows t and t'. Such a time-shifting gap is often caused by time series fluctuations induced by periodicity, seasonality or miscellaneous factors such as weather variation or events. We further discovered that this gap is usually accumulative. A longer temporal interval between two timestamps causes a larger divergence between two distributions. Due to this problem, machine learning models for time series prediction tasks expire frequently. Improving model generality makes the model more robust and avoids fitting to local time series fluctuations.

We propose several graph interaction techniques to address the above problem by enhancing the learning of multi-modal coordinated representations and reinforcing the model performance. Yosinski et al. [8] studied feature transferability in deep learning. It shows that features in lower layers are more general and those in higher layers are more specific. According to this phenomenon, we designed two kinds of graph interaction mechanisms correspondingly for lower layers and higher layers.

In lower layers, input spatiotemporal signal maintains its physical properties as engineered features. According to the case in Figure 1, generating latent features via

compound graph connectivity makes great sense in terms of spatial feature extraction. For lower layer spatial feature extraction, we designed grouped GCN (GGCN), which enables random walk graph convolution on compound graph connectivity. The objective of GGCN is to produce a more abstract multi-modal latent feature representation based on graph convolution operations. This technique addresses the first problem on completeness in spatial feature extraction.

Higher layer features provide high level abstractions for the input signal. It becomes meaningless to explicitly extract feature from a certain region. Leveraging some advances from multi-task learning [9], we adapt multi-linear relationship learning [10] to graph convolution networks and try to find shared information among modality-specific representations. According to characteristics in GCNs, we propose multi-linear relationship GCN (MRGCN), which imposes a tensor normal distribution as the prior distribution of multi-modality graph convolution kernels to learn an explainable, robust and fine-grained relationship among modalities. To further enhance the model generality, we propose to freeze part of the covariance structure in the covariance update algorithm in order to improve output feature independency and alleviate the feature co-adaptation problem. The proposed model generates more general high level feature abstractions. This technique also reduces model training time.

On real-world ride-hailing demand data, our model outperforms state-of-the art baselines by a significant margin. Leveraging the advantage of multi-modal and multi-task learning, our model requires less data and time to reach a low prediction error. In summary, this paper makes the following contributions:

- We propose grouped GCN to produce a compound graph connectivity on a multi-modality graph representation. It makes spatial feature extraction on GCN more complete in urban computing.
- We propose multi-linear relationship GCN to learn better coordinated representations among modalities. It improves the generality for high level abstractions.
- We conduct experiments on two large-scale real-world datasets. The proposed approach achieves more than a 10% error reduction over state-of-the-art baseline methods for ride-hailing demand forecasting.

2. Related Work

Region-level prediction in urban computing. Region-level prediction is a fundamental task in data-driven urban management. There is a rich amount of topics, including citizen flow prediction [11–13], traffic demand prediction [3,14,15], arrival time estimation [16] and meteorology forecasting [17,18]. For these topics, the region-wise relationships are measured as geographical distance. The spatial structures for these prediction tasks are formulated as regular graphs, which are inherently Euclidean structures. Convolution neural network-based models are used for effective prediction.

Non-Euclidean structures exist in station-based prediction tasks, including bike-flow prediction [19], traffic volume prediction [6,7,20] and point-based taxi demand prediction [4]. The spatial structures for these problems are no longer regular. Graph convolution networks are usually leveraged for spatial feature extraction in these tasks. Non-Euclidean structures also exist when incorporating auxiliary data to model region-wise relationships. Yao et al. [15] encoded a region-wise relationship as a graph and used graph embedding as external features for convolution neural networks. Geng et al. [5] used MGCN to model region-wise relationships under multiple modalities.

Multi-modality in urban computing. The core issue for multi-modal machine learning is to build models that can process or relate information from multiple modalities [21]. Traditional multi-modal machine learning problems focus on human sensory modalities, including audio–visual speech recognition [22], multi-media analysis [23] and media description [24]. In urban computing, we usually need to harness knowledge from a diverse family of related datasets. Wei et al. [25] first categorized the diversity of urban computing

datasets, such as POI and air quality, as multi-modality and explored feature transferability among different modalities.

Multi-modal fusion is one of the most challenging problems in urban computing. Most existing works incorporate multi-modality auxiliary data as handcrafted features in a straightforward manner. Tong et al. [4] used multi-modality data as input features for a linear regression model. Zhang et al. [2] and Yao et al. [15] concatenated auxiliary data to high level abstractions for region-level spatiotemporal prediction networks.

GCN-based approaches encode multi-modality data as region-wise relationships and perform as a static structure in deep learning. The spatial feature extraction process on GCN is associated with these modalities. According to applications in traffic volume prediction [6] and taxi demand prediction [5], GCNs are effective in spatial feature extraction on spatial-variant modality data. However, all techniques above fail to build a relationship among modalities, which is expected to improve the generality of the learning framework.

Multi-task relationship learning. Multi-task relationship learning is a basic approach for multi-task learning. Zhang and Yeung [26] first proposed a regularized multi-task model MTRL by placing a matrix-variate normal prior on the model parameters:

$$W \sim \mathcal{MN}(\mathbf{0}, \mathbf{\Sigma}_r, \mathbf{\Sigma}_c)$$

where $\mathbf{\Sigma}_r$ and $\mathbf{\Sigma}_c$ are the row and column covariance. Long et al. [10] proposed Multilinear Relationship Network (MR Network), which learns the multilinear relationship on different modes for the joint-task parameter tensor as:

$$W = [W_1; W_2; \ldots; W_t]$$
$$W \sim \mathcal{TN}_{D_f \times D_c \times D_t}(\mathbf{O}, \mathbf{\Sigma}_f, \mathbf{\Sigma}_c, \mathbf{\Sigma}_t)$$

where W refers to the joint weight by concatenating all fully connected weights from all tasks. D_f, D_c and D_t denote the feature dimension, class dimension and task dimension in the joint weight. $\mathbf{\Sigma}_f$, $\mathbf{\Sigma}_c$ and $\mathbf{\Sigma}_t$ represent covariance for each mode. Experiment results showed that imposing a multilinear relationship regularizer on the last few fully connected layers in CNN-like structures increased the feature generality and transferability in task-specific layers.

However, MR Networks only learn multilinear relationships on fully connected layers. Other deep learning structures, such as CNN or GCN, have more complicated physical meanings.

3. Methodology

$\mathbb{A} = \{A_0, A_1, \ldots, A_{|M|}\}$ denotes adjacency matrices for different graphs. Each graph corresponds to one of the $|M|$ modalities. In the ride-hailing demand prediction problem, each graph represents a kind of pair-wise spatial relationship for regions, including neighborhood (geo-distance) A_N, POI similarity A_S and road connectivity A_C [5].

$$A_{N,i,j} = \begin{cases} 1, \text{if region } i \text{ and } j \text{ are adjacent} \\ 0, \text{otherwise} \end{cases}$$
$$A_{S,i,j} = \text{sim}(P_{v_i}, P_{v_j})$$
$$A_{C,i,j} = \max(0, \text{conn}(v_i, v_j) - A_{N,i,j})$$

A_N defines the adjacency relationship between regions. We construct A_N by connecting a vertex to its 8 neighbors in a 3×3 grid. A_S is the cosine similarity between POI vectors of two regions. Each entry in the POI vector represents the number of POIs in a specific category. A_C indicates the connectivity between two regions. Two regions are connected as long as there is a highway or subway that directly connects them.

We define the one-step spatiotemporal prediction task for a certain modality (graph A_i) on a spatiotemporal observation x as:

$$x_t = G(x_{t-1}, x_{t-2}, \ldots, x_{t-k}; A_i) \tag{1}$$

where G represents any random walk-based graph convolution network. $x_t \in \mathbb{R}^{|V|}$ is the temporal slice of a spatiotemporal observation at time t.

When the graph convolution operation $G : (\mathbb{R}^{|V|\times f_1}; \mathbb{R}^{|V|\times|V|}) \to \mathbb{R}^{|V|\times f_2}$ is defined as the polynomial of the graph laplacian (in this work, we use a symmetric normalized laplacian: $L = I - D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$) L with degree up to K:

$$G_W(X; A) = \sum_{\alpha=0}^{K} L^{\alpha} X W_{\alpha} \tag{2}$$

the above definition refers to the graph convolution operation of ChebNet [27]. In this work, we use this variation of graph convolution operations.

In the multi-modality formulation of this problem, each modality refers to a representation learning process of the same spatiotemporal observation on different graphs. Following the convention in [21], we formally joined a representation of a multi-modality learning problem on a multi-graph convolution network as:

$$x_t = \mathcal{F}_{A_i \in \mathbb{A}}(G_W(x_{t-1}, x_{t-2}, \ldots, x_{t-k}; A_i)) \tag{3}$$

where $\mathcal{F}_{A\in\mathbb{A}}$ denotes the interaction function across multi-graphs. In previous work [5], it is defined as a stacking function in anterior layers and sum function in the output layer. The major contribution of this work focuses on the design of this interaction function.

Figure 2 shows the proposed framework. According to an analysis on feature generality [8] for deep neural networks, we proposed two techniques for building modality-wise interactions targeted for lower layers and higher layers, respectively, in stacked MGCNs. In lower layers, the hidden features are concrete. The feature extraction in lower layers is usually general. Considering these facts, we propose to build inter-modality connections to enable inter-graph spatial feature extraction. To distinguish feature extraction parameters, we penalize inter-graph weight and intra-graph weight differently by group regularization. In higher layers, the hidden features are highly abstract that they can no longer maintain their physical properties. Applying inter-modality connections is not applicable. High level features are usually task specific, which is harmful to model generality and transferability. In these layers, we propose to learn a multilinear relationship on training parameters of joint modalities in order to improve the model generality and avoid overfitting the model to local fluctuations.

3.1. Grouped GCN

Figure 3 shows one layer transformation of grouped GCN (GGCN). In lower layers, we use GGCN to build compound graph connectivity, which enables cross-graph spatial feature extraction.

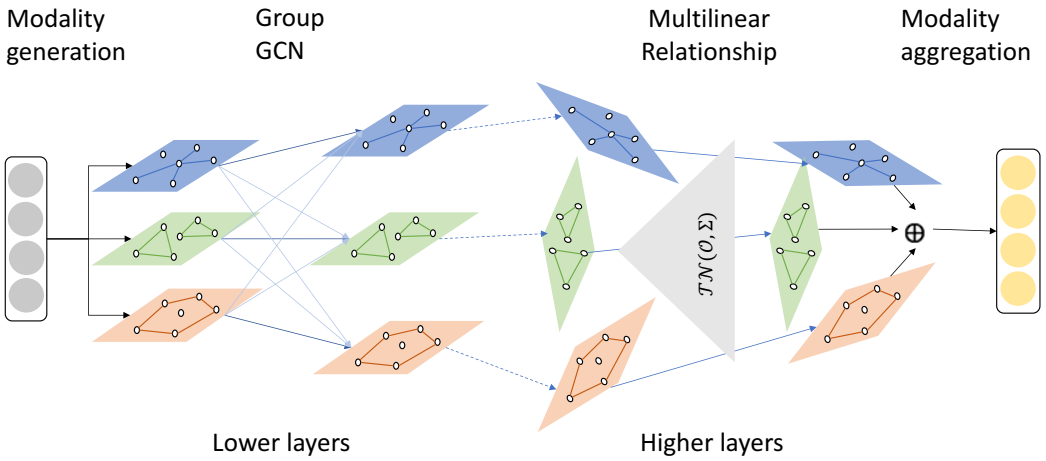

Figure 2. Overview of the proposed graph interaction mechanism for stacked MGCNs. The multi-modality representation of input signals is generated by multi-graphs. In lower layers of deep neural networks, we use grouped GCN to enable inter-graph spatial feature extraction. In higher layers, we use multi-linear relationship GCN to learn a modality-wise relationship by imposing a tensor normal distribution on the joint representation of parameters. Finally, we aggregate modalities to produce an output.

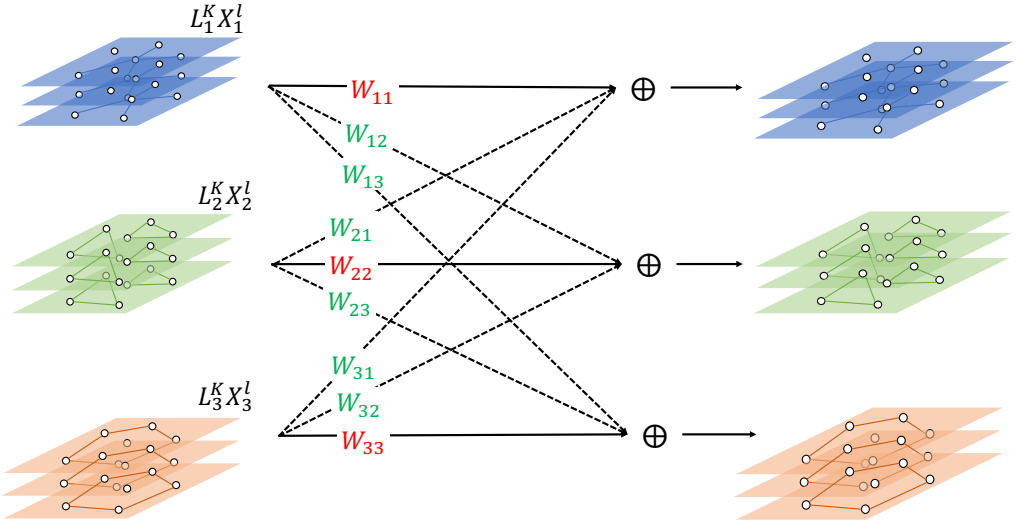

Figure 3. One layer transformation for grouped GCN. Weights marked in red represent intra-modality weights. Green ones represent inter-modality weights.

$L_i \in \mathbb{R}^{|V|\times|V|}$ denotes the graph laplacian matrix of the i-th modality. $X_i^l \in \mathbb{R}^{|V|\times f_l}$ denotes the input signal of the ith modality of the lth layer ($l, i \in \mathbb{Z}^+$). When $l = 1$, X_i^1 represents the raw input and $X_i^1 = X_j^1, \forall i, j$, we define the lth layer parameter W^l as:

$$W^l = \begin{pmatrix} w_{1,1}^l & w_{1,2}^l & \cdots & w_{1,|M|}^l \\ w_{2,1}^l & w_{2,2}^l & \cdots & w_{2,|M|}^l \\ \cdots & \cdots & \cdots & \cdots \\ w_{|M|,1}^l & w_{|M|,2}^l & \cdots & w_{|M|,|M|}^l \end{pmatrix}, \tag{4}$$

$w_{i,j}^l \in \mathbb{R}^{f_l \times f_{l+1} \times K}$ denotes the weight matrix to transform the ith modality input to the jth modality output via ChebNet transformation $G_{w_{i,j}^l}(X_i^l; A_i)$, where f_l and f_{l+1} are the

feature dimension of the lth and $(l+1)$th layer. K represents the degree of the Chebyshev polynomial, which is sliced during the computation of ChebNet.

The jth modality output is computed as:

$$X_j^{l+1} = \sigma(\sum_{i=1}^{|M|} G_{w_{i,j}^l}(X_i^l; A_i) + b_j^l) \tag{5}$$

We denote all weights that transform an input to output within sthe ame modality, i.e., $w_{i,j}^l$ for $\forall i = j$, as intra-modality weights. Similarly, we define the inter-modality weight as $w_{i,j}^l$ for $\forall i \neq j$. It is obvious that when all inter-modality weights are set to 0, the graph convolution operation defined above degrades to MGCN.

Adding cross-modality weights as stated above introduces a tremendous increment on the number of parameters with a factor of $O(|M|)$. This may boost the model complexity and cause overfitting. To address this issue, we used grouped sparsity [28,29] to regularize the complexity of parameters. We designed flexible group regularization loss for layer l:

$$J_1^l = \alpha \sum_{i=j} ||w_{i,j}^l|| + \sum_{i \neq j} ||w_{i,j}^l|| \tag{6}$$

Different from traditional group regularization, we use a tunable parameter α to control the trade-off on penalties for intra-modality weights and inter-modality weights. To maintain the difference among modalities, we prefer a smaller α value in order to introduce less penalty to intra-modality weight. The inter-modality feature extraction focuses on those highly strong relationships. This will help to maintain model generality from multi-modality throughout the proposed GGCN architecture.

The design strategy has several properties that maintain the advantage of GCN models. Firstly, the increment for computational complexity for GGCN is limited. The factor of time complexity increment is $O(M)$, which is the polynomial of the number of modalities. In practice, the number of modalities are usually not large. Secondly, the extra computation above to compute intra-modality transformation and inter-modality transformation are naturally independent. It is easy to design a parallel implementation. Finally, GGCN is a linear combination of different graph laplacians, which keep the numerical stability of the original MGCN model when using the normalized symmetric laplacian.

3.2. Multi-Linear Relationship GCN

In high level layers, latent features no longer maintain their properties as spatiotemporal observations. Instead of building cross-modality connections, we propose to learn multi-linear relationships (MR) on joint-modality weights (we only keep intra-modality weights in high level layers) by imposing a tensor normal distribution as the prior distribution.

The dimensionality transformation of graph convolution operations in ChebNet is shown in Figure 4. There are five dimensions in the whole system in total, including regions/vertices (R, $|R| = |V|$), inputs (I), outputs (O), Chebyshev Polynomial (C, $|C| = K$) and modalities (M). For each single modality task, the representation of input signals on graph laplacian L_i is in a three-dimensional space of region, input and Chebyshev polynomial: $\{L_i^\alpha X | \alpha = 0, 1, \ldots, K\} \in \mathbb{R}^{|V| \times |I| \times K}$. The model parameter for the ith modality is in a three-dimensional space of input, output and Chebyshev polynomial: $W_i^l = \{w_{i,\alpha}^l | \alpha = 0, 1, \ldots, K\} \in \mathbb{R}^{|I| \times |O| \times K}$. The joint representation for multi-modality weight is defined as a four order tensor

$$W^l = [W_1^l, W_2^l, \ldots, W_{|M|}^l] \in \mathbb{R}^{|I| \times |O| \times K \times |M|}$$

Firstly, we impose a tensor normal distribution as prior distribution for W^l

$$W^l \sim \mathcal{TN}_{|I| \times |O| \times K \times |M|}(\mathcal{M}^l, \Sigma^l) \tag{7}$$

where $\mathcal{M}^l$ is the mean tensor. $\Sigma^l = \Sigma^l_I \otimes \Sigma^l_O \otimes \Sigma^l_C \otimes \Sigma^l_M$ is the Kronecker decomposable covariance structure. The density function is estimated as:

$$p(W^l) = 2\pi^{-\frac{\prod_{k=1}^4 d_k}{2}} \left(\prod_{k=1}^4 |\Sigma_k|^{-\frac{\prod_{k=1}^4 d_k}{2d_k}}\right) \times e^{-\frac{1}{2}(W^l - \mathcal{M}^l)^T \Sigma^{-1} (W^l - \mathcal{M}^l)} \tag{8}$$

where $d = [|I|, |O|, K, |M|]$ and represents dimensions for each mode, $\Sigma = [\Sigma^l_I, \Sigma^l_O, \Sigma^l_C, \Sigma^l_M]$. $|\cdot|$ represents the determinant. According to Long et al. [10], for Maximum-a-Posteriori (MAP) estimation for model parameters, learning the posterior distribution of W^l given training data (X, Y) is equivalent to minimizing the negative logarithm for the density of $\prod_l P(W^l)$, where (ignores terms irrelevant to W^l because they have no gradient during back propagation):

$$J^l_2 = \frac{1}{2}(vec(W^l)^T (\Sigma^l)^{-1} vec(W^l)) \tag{9}$$

where $vec(\cdot)$ is the flattening operation to transform a high-dimensional tensor to a 1-d vector. The flip-flop algorithm for updating the covariance matrix of a certain mode Σ_i is:

$$\Sigma^l_{d_i} = \frac{d_i}{\prod_{k=1}^4 d_k} (W^l)_{(i)} (\otimes_{k \neq i} \Sigma_k) (W^l)^T_{(i)} + \epsilon I_{d_i} \tag{10}$$

where ϵI_{d_i} is a trade-off term for numerical stability. $(W^l)_{(i)}$ is the vectorization along the ith mode. Such operation outputs a matrix of shape $\mathbb{R}^{(d_i) \times (\prod_{k \neq i} d_k)}$

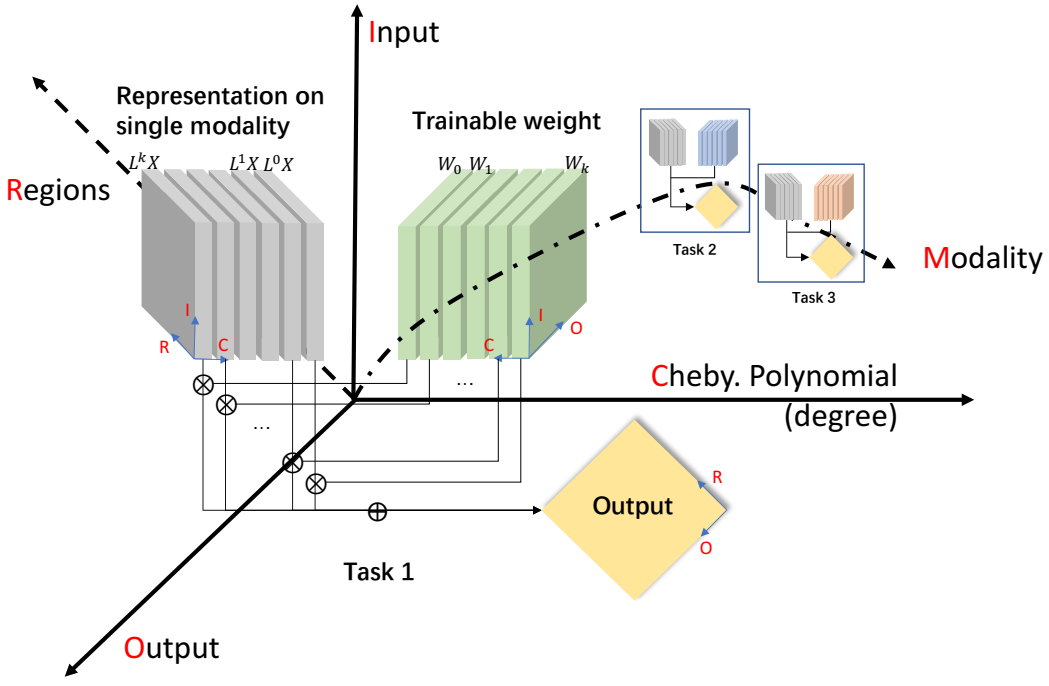

Figure 4. The dimensionality transformation for graph convolution operations in MGCN. Single modality GCN slices input and weight on the *C* mode, multiplies slice pairs and sums up the product.

We further discovered that the covariance update rule above should not be applied to input (I) and output (O) modes. Instead, freezing the covariance matrix of input (I) and output (O) mode to the identity matrix I_d will improve model generality and transferability.

Observing Equation (2) of ChebNet on the lth layer:

$$G_W(X^l; A) = \sum_{\alpha=0}^{K} L^\alpha X^l W_\alpha \tag{11}$$

where $L^\alpha X^l \in \mathbb{R}^{|V| \times f_1}$, $W \in \mathbb{R}^{f_1 \times f_2}$, usually $|V| >> f_1 > f_2$ is due to two facts:

- $|V|$ is very large. $L^\alpha X^l$ is usually sparse.
- In higher layers of DNNs, the feature dimension is usually decreasing, i.e., $f_1 > f_2$.

According to lemma on matrix multiplication:

Lemma 1. *For matrix multiplication* $B = AW$, $Rank(B) \leq \min\{Rank(A), Rank(W)\}$.

The rank for GCN output feature matrix is bounded:

$$\begin{aligned} &Rank(G_W(X^l; A)) \leq \min\{Rank_{|V|}(L^\alpha X^l), \\ &Rank_{|f_1|}(L^\alpha X^l), Rank_{|f_1|}(W_\alpha), Rank_{|f_2|}(W_\alpha)\} \end{aligned}$$

where $Rank_{|f_2|}(W_\alpha)$ is the rank on the f_2 mode of matrix W_α. Increasing the $Rank(W_\alpha)$ on both modes (f_1 and f_2) will lift the upper bound of the output rank. It is known that the co-adaptation problem [30] limits the generality and transferability of DNNs. Initializing and freezing the covariance matrix along the input and output dimension to I_I and I_O will induce a high rank matrix W_α, which lifts the upper bound of rank of output features. The inter-neuron dependency is smaller for a high rank output feature matrix, so that the co-adaptation problem is alleviated and model generality is increased.

3.3. Multi-Modality Fusion

The final layer is the modality fusion layer in order to aggregate features from different modalities and output a prediction result. For a one-step spatiotemporal prediction problem, the output shape is $R^{|V| \times 1}$. The design of the modality fusion is straightforward. First, we make sure the last MRGCN layer reduces the feature dimension to 1. Then, the modality fusion layer is designed as a modality-wise average:

$$O^{l+1} = \frac{1}{|M|} \sum_{j=1}^{|M|} X_j^l, X_j^l \in \mathbb{R}^{|V| \times 1}$$

3.4. Training Algorithm

We combine all loss functions and summarize it for the entire network:

$$\begin{aligned} J_W(X, Y) &= \sum_{s \in S} J_0(f_W(x_s), y_s) + \alpha_{low} \sum_{l \in L_{low}} J_1^l + \alpha_{high} \sum_{l \in L_{high}} J_2^l \\ &= \sum_{s \in S} \frac{1}{|S|} ||f_W(x_s) - y_s||_2 \\ &+ \alpha_{low} \sum_{l \in L_{low}} (\alpha \sum_{i=j}^{i,j=1,..,|M|} ||W_{i,j}^l|| + \sum_{i \neq j}^{i,j=1,..,|M|} ||W_{i,j}^l||) \\ &+ \frac{\alpha_{high}}{2} \sum_{l \in L_{high}} (vec(W^l)^T (\Sigma^l)^{-1} vec(W^l)) \end{aligned}$$

where the J_0 term is the prediction loss of the model. In this work, we use the rooted mean squared error (RMSE) to measure distance between the predicted value and true value. In stacked MGCNs, we set $1, 2, \ldots, l_k$-th layers to L_{low} and use GGCN to construct graph interactions. The remaining layers $l_k, l_k + 1, \ldots$ are set to learn multilinear relationships by MRGCN. The J_1 terms are the GGCN regularizer for each lower layer. The J_2 terms are the

relationship regularizer for MRGCN in the higher layers. α_{low} and α_{high} are the trade-off parameters for regularizers.

The overall training algorithm for the entire network, including GGCN and MRGCN, is shown below (Algorithm 1).

Algorithm 1 Training algorithm for GCN with interactions

Set layers $L_{low} = \{1, 2, \ldots, l_k\}$ to grouped GCN
Set layers $L_{high} = \{l_{k+1}, \ldots\}$ to multi-linear relationship GCN
Initialize $\Sigma_d^l = I_d, \forall l \in L_{high}$ and $d \in \{|I|, |O|, |C|, |M|\}$
Initialize all weights
repeat
  Extract (x_i, y_i) from training set as current training batch
  Update model parameter W according to $J_W(x_i, y_i)$
  Update covariance matrices Σ_C^l and Σ_M^l, $\forall l \in L_{high}$
  Evaluate current model using the validation set
until converge (validation error no longer decreases for several epochs)

4. Experiments

In this section, we compare our graph interaction techniques with state-of-the-art baselines on region-level demand forecasting for ride-hailing service.

4.1. Dataset

We conducted our experiments on two real-world, large-scale ride-hailing datasets collected in two cities: Beijing and Shanghai. Both of the datasets were collected in the main city zone in 2017. We split data to training set (1 March to 31 July 2017), validation set (1 August to 31 October 2017) and test set (1 November to 31 December 2017). The POI data used for A_S contains 13 primary categories, including business building, residential building, entertainments, etc. The road network data used for A_C is extracted from railway, highway and subway dataset from OpenStreetMap [31].

4.2. Experiment Setting

The ride-hailing forecasting problem is a one-step spatiotemporal prediction problem to learn predictor $f : \mathbb{R}^{|V| \times T} \rightarrow \mathbb{R}^{|V| \times 1}$. According to previous works [2,3,5], we set T to 5. Physically, it means predicting the ride-hailing demand in the next time interval using the most recent three ones (closeness), the one in the same time yesterday (period) and the one in the same time last week (trend) [1]. V is the set of regions acquired by partitioning the main city zone to 1 km $\times$ 1 km rectangular grids. Under this setting, there are a total of 1296 regions in Beijing and 896 regions in Shanghai. We set 30 min as the time interval for both training data and test data. Each entry in the spatiotemporal tensor represents the number of ride-hailing demand of a certain region in 30 min.

We propose a 4-layer MGCN, where the first two layers are GGCN and the last two layers are MRGCN. The output dimensions for these layers were set to 32, 64, 32, 1. For all graph convolution operations, the max Chebyshev polynomial K was set to 4. In GGCN, the tunable α was set to 0.1 to maintain intra-modality properties. In MRGCN, the trade-off parameter ϵ was set to 1×10^{-6}. We monitored RMSE on the validation set with early stopping. The regularizers α_{low} and α_{high} were both set to 1×10^{-4}. The neural network was implemented using tensorflow [32] and optimized using adam optimizer [33], with the learning rate as 5×10^{-4} and the batch size as 32. All experiments were conducted in an environment with 10 GB RAM and 9 GB GPU memory of Tesla P40.

4.3. Performance Comparison

Table 1 shows experiment comparisons between the proposed methodology, variations and baselines:

- MGCN: Use one separate GCN to learn prediction task in each modality. There is no graph interaction among modalities.
- STMGCN [5]: Use RNN-based model to extract temporal features ahead of MGCN.
- Share weight: A common technique in multi-task learning. The GCN weight is shared across modalities in each layer.
- Domain adaptation network (DAN) [34]: Minimizing modality divergence by minimizing cross-modality feature divergence. The divergence used is mean maximum discrepancy (MMD).
- $MRGCN_{4\Sigma}$: The proposed multi-linear relationship GCN with all four covariance matrices updated.
- $MRGCN_{2\Sigma}$: Proposed method to freeze covariance matrices for input and output coordinates.

Table 1. Experiment performance in Beijing and Shanghai. The proposed approach achieves the best result among all methods.

Method		RMSE in Beijing	RMSE in Shanghai
Lower Layers	Higher Layers		
STMGCN		10.78	8.30
MGCN		11.82	8.64
GGCN		9.51	8.18
$MRGCN_{2\Sigma}$		9.68	8.30
GGCN	Share weight	9.59	8.13
GGCN	DAN	9.48	8.02
GGCN	$MRGCN_{4\Sigma}$	9.47	7.92
GGCN	$MRGCN_{2\Sigma}$	**9.31**	**7.88**

All proposed methods above are 4-layer MGCNs, with similar hidden feature sizes and same training configurations (learning rate, batch size, etc). We evaluated the model performance according to the prediction error (RMSE) on the test set. The epoch of converge shown in Table 2 and the Table 3 measures the time consumption for each model to reach its optima. Different models converged to different optima. Achieving a lower error usually costs longer training time. We set the benchmark to 10.78 in Beijing, which is the performance of baseline [5] on the same dataset.

Table 2. Number of epochs required to converge to optima or benchmark. The multi-task-based method reduced training time by at least 50%. The experiment was performed in the Beijing dataset. The best results were shown in bold.

Method		Epoch of Converge	Epoch to Break 10.78
Lower Layers	Higher Layers		
STMGCN		115	115
MGCN		110	-
GGCN		130	55
$MRGCN_{2\Sigma}$		95	32
GGCN	Share weight	78	32
GGCN	DAN	72	**24**
GGCN	$MRGCN_{4\Sigma}$	**51**	25
GGCN	$MRGCN_{2\Sigma}$	82	27

Table 3. Training speed for each model to achieve best performance in the ride-hailing demand forecasting task.

	STMGCN	GGCN	MRGCN	GGCN + MRGCN
Min training length	5 months	5 months	3 months	3 months
Training time	110 min	130 min	45 min	60 min

The experiments showed the following facts. Firstly, according to the performance of GGCN, it improved the prediction accuracy for MGCN by invoking more complexity in spatial feature extraction on graphs. With the help of intra-modality transformations, spatial feature extraction was more complete and the model was more expressive. The performance improved by GGCN was even more significant than incorporating an RNN-based temporal feature extraction process (STMGCN). However, with the increment of the parameter size, the model was more prone to overfitting and required longer training time.

Secondly, MRGCN also improved model performance. Compared with GGCN, the influence on prediction error was slightly inferior. There is no significant difference in model capacity and model structure between MRGCN and MGCN. We infer that the multi-linear relationship approach improves prediction performance by improving model generality, so that $MRGCN_{2\Sigma}$ is less prone to overfit to the local fluctuations in the training set and overcomes the gap between the training set and test set. Multi-task learning based approaches, including share weight, DAN and MRGCN, all shortened the model training time. Among these approaches, the share weight method reduced model complexity by a factor of $O(|M|)$, which brought down the prediction performance. The performance of MRGCN and DAN were almost the same.

Thirdly, we show that freezing input and output coordinates in MRGCN is effective. Compared with $MRGCN_{4\Sigma}$, $MRGCN_{2\Sigma}$ decreases the prediction error. This validates our assumption that freezing the covariance for input and output dimension on the weight tensor may induce higher independency among neurons, which alleviates the co-adaptation problem, and thus improves model generality.

Training speed is another important factor to evaluate machine learning models. Table 3 shows the training time required to achieve the optimal performance of each model. We used the grid search to determine the minimum training length of each model. Given a larger training set than this, the model could not converge to a significantly lower validation error. Compared with the baseline, the proposed method reduces the amount of training set and the length of training time by approximately 50%. Among all tested approaches, $MRGCN_{2\Sigma}$ achieved the lowest prediction error on average and on test data after the 4th week. This is an important feature for industrial use. The life cycle for a more generalized model is longer, which reduces the frequency for a model update.

4.4. Model Generality

Figure 5 shows the generalization ability for different models, which validates the above arguments in detail. The data relative divergence (blue bar) was computed as the Kullback Leibler divergence [35] between the temporal pattern of the last week in the training set and temporal patterns of each week in the test set. We discovered that the gap between the training set and test set was accumulative. This indicates that the test data will become more and more divergent from the training data with time shifting. Models are expected to be more general to overcome this phenomenon. According to prediction error by weeks, the prediction error for STMGCN (the gray line) keeps increasing as the test data becomes more divergent. We believe this phenomenon is not caused by model capability, but model generality. For methods including GGCN and MRGCN, the model performance was less influenced by this generalization gap. There was no difference between the model capacity of STMGCN (MGCN) and MRGCN. The network architecture and connectivity were almost the same. This shows that MRGCN has a better generalization ability to avoid overfitting to the training set.

Figure 6 shows the feature inter-dependency of different models. The feature covariance is calculated as the negative logarithm of L2-norm of the covariance matrix along the feature mode. Feature covariance measures the inter-dependency between different neurons in a hidden layer of deep neural network. A higher value represents a lower absolute value for covariance between neurons and a higher neuron dependency. According to the above plot, the neuron independency could be greatly improved by MRGCN. According to Yosinski et al. [8], co-adapted neurons are the major cause for optimization

difficulty in middle layers. Compared with baseline methods, the proposed $MRGCN_{2\Sigma}$ successfully reduced the coherence among hidden layer units and improved generality and transferability for deep neural networks.

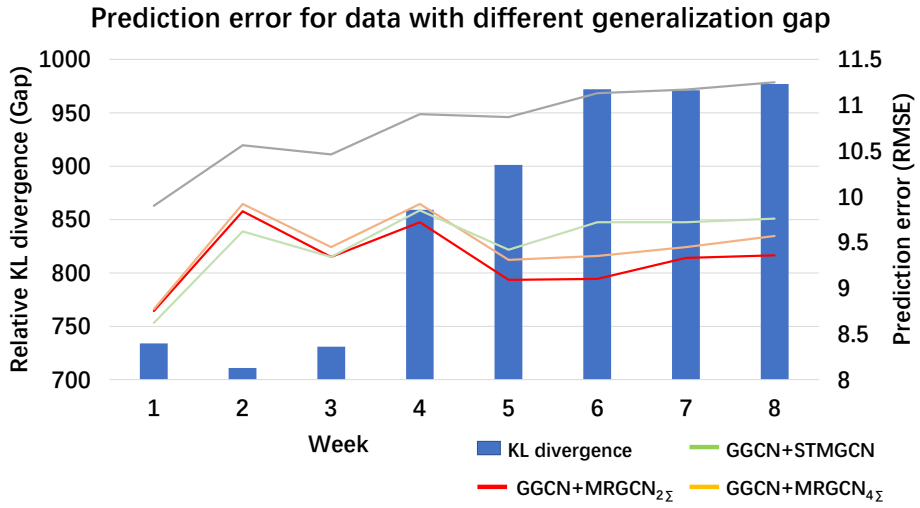

Figure 5. The experiment to test model generality to overcome divergence in the temporal data. The relative data divergence in the test set accumulates along with time. Multi-task learning-based approaches maintain a low prediction error when the data divergence is large.

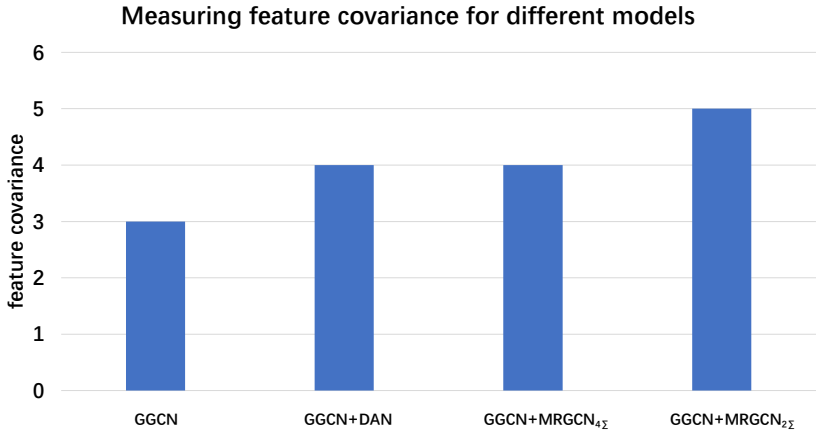

Figure 6. Feature dimension-wise covariance for different models. It is calculated as the negative logarithm of the L2-norm of the covariance matrix of latent features along the feature mode. A higher value indicates higher feature independence.

4.5. Modality Relationship

MRGCN learns explainable relationships between modalities by maintaining a modality-wise covariance matrix. In this part, we first show that all modalities are helpful to the learning task. Then, we will explore the relationship between the modality-wise relationship learnt from optimization and relationship between graphs.

Figure 7 is the Hinton diagram showing the modality-wise relationships for the 3rd and 4th layers in GGCN+ $MRGCN_{2\Sigma}$. N, P, R represent modality for Neighborhood A_N, POI similarity A_S and road connectivity A_C. Similar to the interpretation by [10], we could draw several conclusions. Firstly, most of the tasks are positively correlated (green), implying that all modalities could reinforce the learning of others. This conclusion reaches a consensus with ablation study [5] in Table 4.

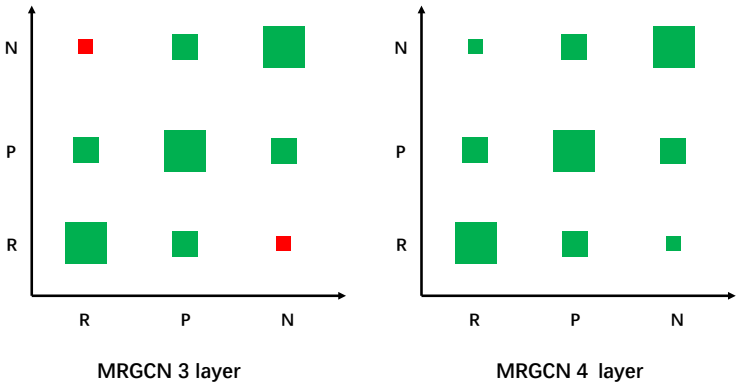

Figure 7. Hinton diagram for modality relationships. The magnitude for relationship is represented by the rectangle size. A green rectangle represents a positive relationship. A red rectangle represents a negative relationship.

Table 4. Ablation study for ST-MGCN. Removing any one modality will result in great damage to the prediction accuracy.

Removed Component	RMSE
Neighborhood	11.47
POI similarity	11.42
Road connectivity	11.69
ST-MGCN	10.78

Secondly, we discovered that the relationship between N and R was weak and random. These two tasks were seemingly related. Compared with that, the relationship R-P and N-P were stable and robust. We try to explain this phenomenon by comparing the graphs A_N, A_S and A_C.

Table 5 shows the density of each graph, which measures the connectivity of the graph in each modality. According to the graph definition, A_S is defined as POI similarity between any region pair, which induces a dense adjacency matrix. A_N and A_C are sparse. We measured the graph similarity by F-measure and edited distance in Table 6. According to the graph definition, edges in A_N are all removed from A_C, so that the edge set $E_N \bigcap E_C = \emptyset$. From the view of graph connectivity, the prediction task on these modalities were hardly related. The relationship A_S-A_C and relationship A_S-A_N were quite similar because A_S was dense. The analysis above helps to understand Figure 7. The relationship between neighborhood (N) and road connectivity (R) was quite random due to the inherent independency between these two modalities. MRGCN learns similar modality relationships for similar graph-pairs. The relationship N-P and R-P were maintained to be similar in both layers.

Table 5. The density of each graph. The graphs are undirected. Density is calculated as $2|E|/|V|(|V|-1)$.

A_N	A_S	A_C
1.3×10^{-3}	1.4	1.4×10^{-3}

Table 6. Two measurements to show similarity between different graphs. F-measurement considers matched and unmatched edges proportional to graph size. Edit distance measures difference between two edge sets.

	A_N-A_S	A_S-A_C	A_C-A_N
F-measure	0.15	0.17	0
Edit distance	1.1×10^6	1.1×10^6	8.8×10^2

5. Conclusions and Future Work

In this work, we propose two graph interaction techniques for multi-modal multi-graph convolution networks. We use GGCN in lower layers to complete graph connectivity for better spatial feature extraction by graph convolution networks. In higher layers, we use MRGCN to learn robust modality relationships. MRGCN alleviates the co-adaptation problem by lifting the upper bound for feature dependency and thus improves the model generality. The experiment on ride-hailing demand prediction shows that our proposed model outperforms baselines in effectiveness, efficiency and robustness. For future work, we plan to investigate the following aspects: (1) evaluate the model with other spatial temporal prediction tasks and other region-wise relationships and (2) explore the impact of sparse and dense graphs on this framework.

Author Contributions: Conceptualization, L.Z.; methodology, L.Z.; software, X.G.; validation, Z.Q., H.W., X.S., X.W., Y.Z., J.L., G.W. and Y.W.; formal analysis, L.Z., X.G.; investigation, L.Z., X.G.; resources, L.Z., X.G.; data curation, L.Z., X.G.; writing—original draft preparation, L.Z., X.G.; writing—review and editing, G.W. and Y.W.; visualization, G.W. and Y.W.; supervision, G.W. and Y.W.; project administration, X.S., X.W., Y.Z., J.L., G.W. and Y.W. ; funding acquisition, X.S., X.W., Y.Z., J.L., G.W. and Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key Research and Development Program of China under Grant No. 2019YFB1600300.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This paper did not report any data.

Acknowledgments: We are grateful to anonymous reviewers for their helpful comments.

Conflicts of Interest: There is no conflict of interest.

Abbreviations

Notation	Type	Meaning
R/V	set	Set of all regions (vertices)
M	set	Set of all modalities
K	scalar	Degree of chebyshev polynomial
I_d	$\mathbb{R}^{d\times d}$	Identity matrix with row/column size d
A_i	$\mathbb{R}^{\lvert V\rvert\times\lvert V\rvert}$	Adjacency matrix of ith modality
L_i	$\mathbb{R}^{\lvert V\rvert\times\lvert V\rvert}$	Symmetric normalized graph laplacian of ith modality
x_t	$\mathbb{R}^{\lvert V\rvert\times 1}$	A spatiotemporal observation (such as ride-hailing demand) value at time t
X_j^l	$\mathbb{R}^{\lvert V\rvert\times f}$	f-dimensional feature of jth modality on lth layer
O^l	$\mathbb{R}^{\lvert V\rvert\times 1}$	Output layer as the lth layer
σ	function	Activation function
f_1, f_2	scalar	Input feature dimension and output feature dimension
For grouped GCN		
b_j	$\mathbb{R}^{\lvert V\rvert\times f}$	Bias for jth modality
W^l	$\mathbb{R}^{\lvert M\rvert\times\lvert M\rvert\times K\times f_1\times f_2}$	Weight of lth layer
$w_{i,j}^l$	$\mathbb{R}^{K\times f_1\times f_2}$	Weight for transforming X_i^l to X_j^{l+1}
w_α^l	$\mathbb{R}^{f_1\times f_2}$	Weight corresponding to a specific chebyshev polynomial term
For multi-linear relationship GCN		
$\lvert I\rvert, \lvert O\rvert$	scalars	Input and Output dimension used to measure weight dimension
W^l	$\mathbb{R}^{\lvert M\rvert\times K\times f_1\times f_2}$	Weight of lth layer
$W_{i,\alpha}^l$	$\mathbb{R}^{f_1\times f_2}$	Weight of lth layer for ith modality and αth chebyshev polynomial term

d	array of 4	dimension of each mode in Σ
$\Sigma_{d_i}^{l}$	$\mathbb{R}^{d_i \times d_i}$	Covariance for the d_ith mode
Σ^{l}	$\mathbb{R}^{\prod d_i \times \prod d_i}$	Kronecker decomposable covariance structure for tensor normal distribution
L_{high}	set	Higher layers assigned to multi-linear relationship GCN
L_{low}	set	Lower layers assigned to grouped GCN

References

1. Zhang, J.; Zheng, Y.; Qi, D.; Li, R.; Yi, X. DNN-based prediction model for spatio-temporal data. In Proceedings of the SIGSPATIAL, Burlingame, CA, USA, 31 October–3 November 2016; p. 92.
2. Zhang, J.; Zheng, Y.; Qi, D. Deep Spatio-Temporal Residual Networks for Citywide Crowd Flows Prediction. In Proceedings of the AAAI, San Francisco, CA, USA, 4–9 February 2017; pp. 1655–1661.
3. Ke, J.; Zheng, H.; Yang, H.; Chen, X.M. Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach. *Transp. Res. Part C Emerg. Technol.* **2017**, *85*, 591–608. [CrossRef]
4. Tong, Y.; Chen, Y.; Zhou, Z.; Chen, L.; Wang, J.; Yang, Q.; Ye, J.; Lv, W. The simpler the better: A unified approach to predicting original taxi demands based on large-scale online platforms. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; pp. 1653–1662.
5. Geng, X.; Li, Y.; Wang, L.; Zhang, L.; Yang, Q.; Ye, J.; Liu, Y. Spatiotemporal Multi-Graph Convolution Network for Ride-hailing Demand Forecasting. In Proceedings of the 2019 AAAI Conference on Artificial Intelligence (AAAI'19), Atlanta, GA, USA, 8–12 October 2019.
6. Li, Y.; Yu, R.; Shahabi, C.; Liu, Y. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In Proceedings of the International Conference on Learning Representations (ICLR '18), Vancouver, BC, Canada, 30 April–3 May 2018.
7. Yao, H.; Tang, X.; Wei, H.; Zheng, G.; Yu, Y.; Li, Z. Modeling Spatial-Temporal Dynamics for Traffic Prediction. *arXiv* **2018**, arXiv:1803.01254.
8. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 3320–3328.
9. Zhang, Y.; Yang, Q. A survey on multi-task learning. *arXiv* **2017**, arXiv:1707.08114.
10. Long, M.; Cao, Z.; Wang, J.; Philip, S.Y. Learning multiple tasks with multilinear relationship networks. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 1594–1603.
11. Jiang, R.; Yin, D.; Wang, Z.; Wang, Y.; Deng, J.; Liu, H.; Cai, Z.; Deng, J.; Song, X.; Shibasaki, R. Dl-traff: Survey and benchmark of deep learning models for urban traffic prediction. In Proceedings of the 30th ACM international Conference on Information & Knowledge Management, Gold Coast, Australia, 1–5 November 2021; pp. 4515–4525.
12. Wang, Z.; Xia, T.; Jiang, R.; Liu, X.; Kim, K.S.; Song, X.; Shibasaki, R. Forecasting Ambulance Demand with Profiled Human Mobility via Heterogeneous Multi-Graph Neural Networks. In Proceedings of the 2021 IEEE 37th International Conference on Data Engineering (ICDE), Chania, Greece, 19–22 April 2021; pp. 1751–1762.
13. Yin, D.; Jiang, R.; Deng, J.; Li, Y.; Xie, Y.; Wang, Z.; Zhou, Y.; Song, X.; Shang, J.S. MTMGNN: Multi-time multi-graph neural network for metro passenger flow prediction. *Geoinformatica* **2022**, 1–29. [CrossRef]
14. Ke, J.; Yang, H.; Zheng, H.; Chen, X.; Jia, Y.; Gong, P.; Ye, J. Hexagon-Based Convolutional Neural Network for Supply-Demand Forecasting of Ride-Sourcing Services. *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 1460–4173. [CrossRef]
15. Yao, H.; Wu, F.; Ke, J.; Tang, X.; Jia, Y.; Lu, S.; Gong, P.; Ye, J.; Li, Z. Deep Multi-View Spatial-Temporal Network for Taxi Demand Prediction. In Proceedings of the 2018 AAAI Conference on Artificial Intelligence (AAAI'18), New Orleans, LA, USA, 2–7 February 2018.
16. Li, Y.; Fu, K.; Wang, Z.; Shahabi, C.; Ye, J.; Liu, Y. Multi-task Representation Learning for Travel Time Estimation. In Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD '18), London, UK, 19–23 August 2018.
17. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.C. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, USA, 7–12 December 2015; pp. 802–810.
18. Shi, X.; Gao, Z.; Lausen, L.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.C. Deep learning for precipitation nowcasting: A benchmark and a new model. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5617–5627.
19. Chai, D.; Wang, L.; Yang, Q. Bike Flow Prediction with Multi-Graph Convolutional Networks. In Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Seattle, WA, USA, 6–9 November 2018.
20. Yu, B.; Yin, H.; Zhu, Z. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'18), Stockholm, Sweden, 13–19 July 2018.
21. Baltrušaitis, T.; Ahuja, C.; Morency, L.P. Multimodal machine learning: A survey and taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, 41, 423–443. [CrossRef] [PubMed]
22. Yuhas, B.P.; Goldstein, M.H.; Sejnowski, T.J. Integration of acoustic and visual speech signals using neural networks. *IEEE Commun. Mag.* **1989**, *27*, 65–71. [CrossRef]

23. Atrey, P.K.; Hossain, M.A.; El Saddik, A.; Kankanhalli, M.S. Multimodal fusion for multimedia analysis: A survey. *Multimed. Syst.* **2010**, *16*, 345–379. [CrossRef]
24. Hodosh, M.; Young, P.; Hockenmaier, J. Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Intell. Res.* **2013**, *47*, 853–899. [CrossRef]
25. Wei, Y.; Zheng, Y.; Yang, Q. Transfer Knowledge between Cities. In Proceedings of the Knowledge Discovery and Data Mining (KDD), San Francisco, CA, USA, 13–17 August 2016; pp. 1905–1914.
26. Zhang, Y.; Yeung, D.Y. A convex formulation for learning task relationships in multi-task learning. *arXiv* **2012**, arXiv:1203.3536.
27. Defferrard, M.; Bresson, X.; Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 3844–3852.
28. Yuan, M.; Lin, Y. Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser. B (Stat. Methodol.* **2006**, *68*, 49–67. [CrossRef]
29. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B (Methodol.)* **1996**, *58*, 267–288. [CrossRef]
30. Hinton, G.E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R.R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv* **2012**, arXiv:1207.0580.
31. Haklay, M.; Weber, P. Openstreetmap: User-generated street maps. *IEEE Pervasive Comput.* **2008**, *7*, 12–18. [CrossRef]
32. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16), Savannah, GA, USA, 2–4 November 2016.
33. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. In Proceedings of the International Conference on Learning Representations (ICLR '14), San Diego, CA, USA, 7–9 May 2015.
34. Long, M.; Cao, Y.; Wang, J.; Jordan, M.I. Learning transferable features with deep adaptation networks. *arXiv* **2015**, arXiv:1502.02791.
35. Kullback, S.; Leibler, R.A. On information and sufficiency. *Ann. Math. Stat.* **1951**, *22*, 79–86. [CrossRef]