Graph Convolutional Regression Networks for Quantitative Precipitation Estimation

Yajing Wu, Yongqiang Tang, Xuebing Yang, Wensheng Zhang, and Guoping Zhang

Abstract—Accurate and high-resolution quantitative precipitation estimation (QPE) plays a crucial role in meteorology and hydrology. However, for acquiring a more accurate QPE, how to depict the complex nonlinear relationship between the radar reflectivity and the true rain rates, as well as adaptively explore the spatial dependencies of precipitation, remains extremely challenging. In this letter, we propose to incorporate the merits of graph convolutional regression networks (GCRNs) and address the aforementioned issues simultaneously in the GCRNs framework. Furthermore, in order to tolerate the variabilities of spatial correlation in the practical precipitation, we expand GCRNs with a multiconvolutional mechanism between the center node and its neighbor rain gauges. Thus, the ability to capture more complicated spatial characteristics of precipitation can be enhanced, and the phenomenon of overwhelming by the neighbor nodes can be relieved. Extensive experiments were implemented on 12 rainfall processes in Hangzhou, China, 2015. The experimental results confirm that our proposal consistently outperforms the state-of-the-art QPE models.

Index Terms—Deep learning, graph convolutional networks (GCNs), modeling, precipitation estimation.

I. INTRODUCTION

ACCURATE and high-resolution quantitative precipitation estimation (QPE) is crucial to various meteorological and hydrological applications. As an indirect measurement of precipitation density, weather radar has been widely used to acquire the information of rainfall fields with a high spatial resolution [1]. The conventional methods for converting radar reflectivity into rainfall rates highly rely on the Z–R relationship (ZR) [2]. Nevertheless, in such a physical model, there exist two main drawbacks that render the unsatisfactory QPE performance. First, the ZR is subject to strictly ideal relationship (ZR) [2]. Nevertheless, in such a physical model, there exist two main drawbacks that render the unsatisfactory QPE performance. First, the ZR is subject to strictly ideal conditions that are difficult to meet in a practical situation. Second, the general operation of the ZR model treats each grid independently and, therefore, ignores the spatial dependencies of precipitation.

The past several decades have witnessed numerous methods for overcoming the aforementioned issues. For the first issue, various data-driven methods were proposed to capture the complex nonlinear relationship between radar reflectivity and rain gauge observations, such as support vector machine (SVM) [3], artificial neural networks (ANNs) [4], and random forest (RF) [5]. Nevertheless, most of them typically regard the rainfall event as spatially stationary and fail to consider the spatial correlation. To address this issue, geostatistical interpolation methods have been widely employed to merge the information of neighbor rain gauges, such as conditional merging (CM) [6] and cokriging [7]. Unfortunately, the existing geostatistic-based methods generally need to fit a complicated variogram function with Gaussianity assumption, such that their performance is limited by the particular subjective definition of variogram models [8]. Therefore, how to effectively model the complex nonlinear relationship between radar reflectivity and rainfall rates, as well as adaptively explore the spatial dependencies within rainfall fields at the same time, is still a challenging problem.

Recent advances in deep learning, especially convolutional neural networks (CNNs) [17], provide useful insights to the grid-like data (e.g., image). However, in many other contexts, the input data are actually lying on irregular domains, including social networks, protein interfaces, traffic sensor networks, and rain gauges networks. Fortunately, graph convolutional networks (GCNs) fill in this puzzle by generalizing deep learning to the data with arbitrary graph structures and have been applied to many tasks successfully [11]–[13]. Generally, GCNs obtain the representation of a node via fusing the information of neighbor rain gauges, such as conditional merging (CM) [6] and cokriging [7]. Unfortunately, the existing geostatistic-based methods generally need to fit a complicated variogram function with Gaussianity assumption, such that their performance is limited by the particular subjective definition of variogram models [8]. Therefore, how to effectively model the complex nonlinear relationship between radar reflectivity and rainfall rates, as well as adaptively explore the spatial dependencies within rainfall fields at the same time, is still a challenging problem.

In this letter, we propose to formulate the QPE as a graph convolutional regression networks (GCRNs) learning problem to simultaneously model two kinds of relationships: 1) the nonlinear mapping between radar reflectivity and rainfall rate at the point to be estimated and 2) the spatial correlation between the estimated point and neighbor rain gauges. To the best of our knowledge, this is the first work to apply the GCRNs framework to QPE. Furthermore, to enable GCRNs...
tolerating some variabilities of spatial correlation in the practical rainfall fields, a multiconvolutional operation mechanism is proposed to explore more complicated spatial characteristics of precipitation. In virtue of this new mechanism, the phenomenon of overwhelming by the neighbors can be mitigated effectively. When evaluated on real-world precipitation data sets, our proposal consistently outperforms state-of-the-art QPE methods.

II. PROPOSED METHOD

This work focuses on node regression for precipitation estimation, whereas the existing GCNs cannot deal with the QPE directly. Thus, in this section, first, the construction of the graph is presented, followed by details of the basic GCRNs framework. Furthermore, the convolution mechanism is expounded and added to the basic framework.

A. Graph Construction

The goal of QPE on graph is to obtain the continuous precipitation value for the node \( v_i \in \mathcal{V} \), where \( \mathcal{V} \) is the set of \( N \) nodes. In order to train GCRNs with observations by backpropagation, the value \( y_i \) obtained by a rain gauge is regarded as the node’s label in our graph. The GCRNs is trained by minimizing mean absolute error (MAE) loss

\[
\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i|
\]

where \( \hat{y}_i \) is precipitation estimation obtained by GCRNs.

For the node feature \( \mathbf{x}_i \), we typically choose the constant altitude plan position indicator (CAPPi) reflectivities. The spatial interpolation is applied to maintain spatial consistency of radar and rain gauges. Here, we adapt the lowest four levels of CAPPi reflectivities with the nearest nine grid points for each level as in [4] and [15]. Because the node is built on the rain gauge whose location is available by the latitude and longitude coordinates, the nodewise edge is naturally modeled by geographical distance. Thus, we compute the pairwise distances \( \text{dist}(v_i, v_j) \) between nodes \( v_i \) and \( v_j \). Then, we build the edges \( A_{ij} \) with the thresholded Gaussian kernel [16], resulting in the following expression:

\[
A_{ij} = \begin{cases} 
1, & \exp(-\frac{\text{dist}(v_i, v_j)^2}{\sigma^2}) \leq \kappa \\
0, & \text{otherwise}
\end{cases}
\]

where \( \sigma \) is the standard deviation (a constant statistic) of distances and \( \kappa \) is the threshold [13]. Here, the threshold needs to be selected in accordance with specific data. The section about data will illustrate its setting. According to the edges, adjacency matrix \( A \in \mathbb{R}^{N \times N} \) can be obtained. The graph construction in Fig. 2 illustrates an example of the graph for QPE. By converting the rain gauges network into a graph, spatial relationships that exist within the data have been represented explicitly.

B. Basic GCRNs Framework

After finishing graph construction, we present the framework of GCRNs inspired by the classical inductive GCNs [14]. The caption and the top figure of Fig. 2 illustrates the framework of GCRNs. Broadly speaking, the GCRNs framework can be considered as a bottom–up network, where the neighbor feature and structural information are fused layer by layer and, eventually, reach to the point to be estimated (red node). In each layer, the representation \( h_i \) for a node \( v_i \) is updated by the aggregation function (AG) and weight matrices \( \mathbf{W} \) from their own neighbors \( v_j \):

\[
h_i^l \leftarrow \text{ReLU}(\mathbf{W}^{l-1} \cdot \text{AG}(\{h_j^{l-1} \forall v_j \in \mathcal{N}(v_i)\}))
\]

\[
h_i^l \leftarrow h_i^{l-1} / \|h_i^{l-1}\|_2
\]

where \( \mathcal{N}(v_i) \) represents the neighbors of \( v_i \), and the superscript \( l \) is the layer index. The GCRNs fuse the representations of a center node and its neighbors to get a new representation for the center node, also as depicted by the orange and blue areas in Fig. 2. In general, max-pooling or mean-pooling can be used as the AG. Here, our basic graph embedding adopts the max-pooling as follows:

\[
h_i^{l+1} \leftarrow \text{MAX}(\{h_j^{l-1} \forall v_j \in \mathcal{N}(v_i)\})
\]

where MAX represents the elementwise max operator. In (3), GCRNs concatenates (|) the center node’s representation and feed it into a multilayer perceptron (MLP) with the rectified linear unit (ReLU) nonlinearity followed by normalization (4). In practice, before each layer, an MLP is also required to obtain sufficient expressive power to transform both center and neighbor features \( h_i^{l-1} \) into higher level features for regression. The bottom figure of Fig. 2 details the flowchart (see the arrow marked Basic GCRNs).

In this work, following the classical GCNs [14], the GCRNs is stacked by two blocks, each of which contains two layers. When the GCRNs reaches the center nodes \( v_i \in B^L \) in the output layer \( L \) with the minibatch of center nodes \( B^L \), the precipitation values estimated by GCRNs are represented

\[
\hat{y}_i \leftarrow \text{ReLU}(\mathbf{W}^{\text{Out}} \cdot h_i^{L-1}) \quad \forall v_i \in B^L.
\]

C. Enhanced GCRNs With Convolutional Mechanism

Although the aforementioned basic GCRNs can be leveraged for QPE, its performance is still unsatisfactory in the complicated atmospheric situations. Here, an enhanced GCRNs (E-GCRNs) with the multiconvolutional mechanism is proposed to explore the complicated spatial characteristics in an even better fashion. Specifically, we first construct node pairs, such as the patches of CNNs. Each patch in the graph includes the center and a neighbor’s hidden representation. Then, the identical convolutional kernel is shared by each patch in a GCRNs layer. In practice, the patches
Fig. 2. Framework of the proposal, with two main steps: 1) graph construction (pentagram indicates points to be estimated) and 2) GCRNs is leveraged for effective precipitation estimation. During a forward process, the minibatch of center nodes (colored red) $B^t$ is sampled randomly from the training nodes, followed by $B^{t-1}$ that contains their own one-order neighbors (yellow nodes) and $B^{t-2}$ that contains two-order neighbors (green nodes) concerning edges. According to Fig. 1, the same color of the node corresponds to the same order neighbor. In layer 1, one-order neighbors are updated with the information from two-order neighbors. Subsequently, the center node is updated with the information from one-order neighbors in layer 2. Here, the layer can be considered as a subcomponent of the block. After conducting a complete block, the information of the center node will be updated from bottom to up.

Fig. 3. Information about the data set. (a) Characteristics of Hangzhou radar. (b) Hangzhou radar coverage area. (c) Graph construction with $\kappa = 0.4$, which is set to ensure that each node contains at least one neighbor.

are constructed by replicating the center node embedding for each neighbor to perform subsequent convolution operations (illustrated by the bottom figure of Fig. 2). Here, enhanced representation $z_{ji}$ for one patch can be defined via the Hadamard product ($\ast$)

$$z_{ji} = h_{j}^{i} * C_{1,:} + h_{i}^{i} * C_{2,:}.$$  (7)

where $C \in \mathbb{R}^{2 \times M}$ is the convolutional kernel with two-row vectors $C_{1,:}$ and $C_{2,:}$. After moving across all patches, the enhanced representations of neighbor nodes can be obtained by

$$z_{j} = \text{Conv}(h_{j}^{i}, \{h_{j}^{i} \forall v_{j} \in N(v_{i})\}; C).$$  (8)

Here, Conv is the whole convolutional operations.

Furthermore, it is known that more convolutional kernels usually yield more stable and better performance. Similar to [18], we employ multiconvolutional kernels. Specifically, when $K$ independent convolutional operations are executed with (8), the $K$ channels enhanced representations can be obtained. We average the $K$ representations by

$$z_{j} = \frac{1}{K} \sum_{k=1}^{K} \text{Conv}(h_{j}^{i}, \{h_{j}^{i} \forall v_{j} \in N(v_{i})\}; C^{k}).$$  (9)

The proposed multiconvolutional mechanism can be viewed as a ready-to-use plug-in, which can be easily embedded between the MLP and AG in Fig. 2.

III. EXPERIMENTAL STUDY

A. Data Set

The data of the precipitation estimation for this study are provided by the China Meteorological Administration (CMA). The radar reflectivities data are collected by Hangzhou (Zhejiang Province, China) radar with the volume scan interval of 6 min. The 35 rain gauge data used in this study are collected by the Automatic Weather Station (AWS) every minute over the Hangzhou radar coverage area. The characteristics of Hangzhou radar are listed in Fig. 3(a). Fig. 3(b) shows the locations of the rain gauges. The radar reflectivities and the observations were both collected from January to October 2015, including 12 rainfall processes. Besides, rain gauge data are accumulated to 6 min to maintain temporal consistency with radar data. Fig. 3(c) shows the result of graph construction for our data in the same time step. The data contain a total of 950 time steps. To avoid that different splits of data may affect the performance of the GCRNs models, the leave-one-out method is adopted to evaluate our proposal. Especially, for 35 rain gauges in Hangzhou, the GCRNs are trained and tested 35 times independently. During each training and estimation, the nodes belonging to one rain gauge are selected as the test data ($1 \times 950$ nodes). The remaining data are used to train our model and determine the hyperparameters. Besides, the Z-score normalization is employed to preprocess the input features.
B. Competitors and Model Configurations

In this letter, our proposed GCRNs is compared with the following five QPE methods, namely, ZR, fully convolutional networks (FCNs), SVR (SVM for regression), RF, and CM (see Section I).

1) ZR ($Z = aR^b$) [2]: on each training data set, $a$ is varied from 31 to 400 and $b$ from 1.1 to 1.9. The optimal parameters are obtained when the validation reaches minimal MAE.

2) For validating the spatial dependence influence, FCN, which is a submodel of the basic GCRNs, remains the same configuration as the GCRNs except for structural information.

3) For SVR, the radial basis function (RBF) kernel is adopted.

4) We vary the tree number from 100 to 200 with an interval of 10 to determine the optimal parameter for RF.

5) The CM [6] method applies ordinary kriging with an exponential model to obtain deviation between the interpolated and observed radar rainfall. Then, the deviation is used to remove the bias in the rain gauge interpolation field. Among these methods, SVR and RF are considered as conventional nondeep learning methods for comparison.

We first evaluate the basic GCRNs. Following [14] and [19], the grid search is adopted as the hyperparameter selection strategy. For our data set, the following hyperparameters are tuned: 1) learning rate in {0.01, 0.05, 0.001}; 2) batch size in {512, 1024}; 3) hidden dimension of MLP and output dimension in each layer in {256, 512, 1024}; and 4) for epoch, we apply the early stopping criterion with a window size of 30. Batch normalization [20] is also used after every layer in the GCRNs. Meanwhile, 35-fold cross-validation is a heavy workload for hyperparameter selection. Therefore, following [21], we fixed the number of neighbors at 12, which is almost the average number of neighbors in our graph. Second, the E-GCRNs with the optimal three convolutional kernels are also evaluated, whose hyperparameters are kept the same with basic GCRNs to ensure a fair comparison.

Here, four verification scores are used to assess the performance. In detail, they are the MAE (1), root-mean-square error (RMSE), median absolute error (MedianAE), and correlation coefficient (CC). Given the observation $y_i$ and the precipitation estimation $\hat{y}_i$, the equations are defined as

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2}$$

(10)

$$\text{MedianAE} = \text{Median}(|y_i - \hat{y}_i|)$$

(12)

$$\text{CC} = \frac{\text{cov}(y, \hat{y})}{\sigma_y \sigma_{\hat{y}}}$$

(11)

$\sigma_y$ and $\sigma_{\hat{y}}$ are the observations and radar estimation over the entire study.

Table I shows all the evaluation results for the competitors, i.e., the basic GCRNs and E-GCRNs. Compared with ZR, FCN achieves superior performance owing to the extracted more informative features. However, the power of FCN is still limited by only utilizing the nodes’ features without any spatial information. Besides, the experimental results strongly support the advantage of machine learning methods and considering spatial information. In particular, SVR and RF are significantly better than ZR due to the superior capacity of nonlinear modeling. Compared with ZR, the performance gain of CM lies in the utilization of spatial correlation information. Overall, both the basic and our E-GCRNs outperform all the competitors. With the convolutional operations added into the basic GCRNs, the E-GCRNs achieve the best scores among all the methods. Especially, compared with ZR, improvements of 39% in MAE, 39% in MedianAE, 31% in RMSE, and 11% in CC are obtained by E-GCRNs. From ZR to E-GCRNs, the improvement is achieved step-by-step, which can verify the contributions of the two major concerns in our E-GCRNs.

To have a closer look at the performance of the best-performing model, the rain gauge observations, and the estimations obtained by ZR, the basic GCRNs and E-GCRNs at different moments are shown in Fig. 4. Clearly, ZR tends to underestimate the values in most cases, while our basic GCRNs outperform ZR. Some dashed circles are identified to

![](image-url)

Fig. 4. Key component validation: effectiveness of basic GCRNs and E-GCRNs with multiconvolutional mechanism. Rain gauge observations, and the estimations obtained by ZR, the basic GCRNs, and E-GCRNs at different moments are shown in (a), (b), and (c), respectively.
examine the role of the convolution mechanism. Compared with our basic GCRNs, by using the proposed convolution mechanism, the E-GCRNs release the phenomenon of overwhelming by the neighbor nodes. For instance, the estimated value is less influenced by the surrounding neighbors [see Fig. 4(a)]. Moreover, in some more complicated weather conditions [such as in Fig. 4(b) and (c)], E-GCRNs can better capture the complicated spatial correlation. This finding can support the superiority of our E-GCRNs.

To further visually examine the quality of hourly QPEs by each method, the hourly radar rainfall estimations [ZR, SVR, RF, CM, and our best model (i.e., the E-GCRNs)] against gauge observations are compared. Three gauges, corresponding to different rainfall process, are illustrated in Fig. 5. In general, compared with the other four methods, our proposal achieves the best performance, and the hourly total estimates from our proposal (red line) approximately match the observations (black line). Therefore, these visual results can further confirm the effectiveness of our proposal. In addition, we also analyze the effects of the different thresholds \( \kappa \) in (2). We vary \( \kappa \) from 0.4 to 0.8. The performance is improved slightly at \( \kappa = 0.5 \) and then decreased with the increase in the value of the threshold. As the value of the threshold is increased, the neighbors can contain more nodes that are farther from the center node. However, when too far nodes are considered as neighbors, the spatial correlation may be so weak that it brings negative effects to our model.

IV. CONCLUSION

In this letter, we formulate the QPE problem on the graph and build the model with GCRNs framework to simultaneously model the complex nonlinear mapping from radar reflectivities to true rainfall rates and capture the spatial dependencies within rainfall fields. Furthermore, we propose a novel multiconvolutional mechanism between the center node and its neighbor rain gauges to deal with the complicated weather situations. To demonstrate the advantage of our proposal, extensive experiments were implemented on 12 rainfall processes in Hangzhou, China, 2015. The experimental results confirm the superiority of the GCRNs over the state-of-the-art QPE methods. In this letter, we focused on constructing a graph by distance. Future work might explore ways to utilize similar rainfall events to enrich the graph.

REFERENCES