Unsupervised Air Quality Interpolation with Attentive Graph Neural Network

Thu Hang Phung*
Duc Long Nguyen*
hang.pt194758@sis.hust.edu.vn
long.nd183583@sis.hust.edu.vn
Hanoi University of Science and Technology
Hanoi, Vietnam

Viet Hung Vu
Hanoi University of Science and Technology
Hanoi, Vietnam
hung.vv162050@sis.hust.edu.vn

Thanh Trung Huynh
EPFL
Lausanne, Switzerland
thanh.huynh@epfl.ch

Thanh Hung Nguyen†
Hanoi University of Science and Technology
Hanoi, Vietnam
hungnt@soict.hust.edu.vn

Phi Le Nguyen
Hanoi University of Science and Technology
Hanoi, Vietnam
lenp@soict.hust.edu.vn

ABSTRACT

Rapid industrial expansion, urbanization, and traffic growth have led to a decline in air quality that significantly impacts human health and environmental sustainability, particularly in developing nations. Due to the limited number of monitoring stations, the air quality index is not gathered at numerous locations. To address the difficulty of predicting the air quality value at an arbitrary place, several studies, including statistical and machine learning approaches, have been proposed. The majority of existing research employs classic distance-based interpolation techniques. In this paper, we propose a novel attentive neural-based approach for estimating unmonitored air quality values. This method follows the encoder-decoder paradigm, with the encoder and decoder being learned independently utilizing distinct processes. In the encoder, we propose AGE, an inductive unsupervised learning methodology that integrates attention mechanisms. AGE learns a set of functions that generate spatial embeddings by aggregating features from the surrounding region. For the decoder, we utilize the Gated Recurrent Unit and a fully-connected layer to estimate the air quality index at the targeted location. We conduct extensive experiments to evaluate the performance of our proposed method and compare it to the state-of-the-art (SOTA). The experimental results show that our approach reduces the estimation error from 8.07% to 37.04% compared to the SOTA.

*Both authors contributed equally to this research.
†Corresponding author

SoICT 2022, December 1–3, 2022, Hanoi, Vietnam

ACM ISBN 978-1-4503-9725-4/22/12…$15.00
https://doi.org/10.1145/3568562.3568657

CCS CONCEPTS

• **Computing methodologies** → *Knowledge representation and reasoning*; **Artificial intelligence**; **Image segmentation**; **Supervised learning**; • **Computer systems organization** → **Neural networks**.

KEYWORDS

Air quality interpolation, Time series prediction, Machine learning, Deep neural network, Graph neural network

ACM Reference Format:
Thu Hang Phung, Duc Long Nguyen, Viet Hung Vu, Thanh Trung Huynh, Thanh Hung Nguyen, and Phi Le Nguyen. 2022. Unsupervised Air Quality Interpolation with Attentive Graph Neural Network. In *The 11th International Symposium on Information and Communication Technology (SoICT 2022), December 1–3, 2022, Hanoi, Vietnam.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3568562.3568657

1 INTRODUCTION

Background. Air pollution is one of the most worrisome concerns on a worldwide scale since it negatively impacts human health and causes several severe problems. According to the World Health Organization, lung cancer caused by air pollution accounts for 29% of all lung cancer-related fatalities, hence increasing the need for governments and non-governmental organizations to manage and predict air pollution levels [17]. Ideally, we would like a fine-grained map that can offer data on air quality at any arbitrary location. Traditionally, air quality data is collected by monitoring stations placed at fixed locations. Although these stations can gather air quality indices with a high degree of accuracy, their installation and operation costs are so substantial that only a small number of stations are placed, resulting in a relatively sparse network. Consequently, one cannot rely solely on these stationary air quality monitoring sites to achieve a fine-grained air quality map. For this purpose, several attempts have been made to develop a novel method known as *spatial air quality estimation*, which forecasts air quality in unmonitored locations using data collected from adjacent monitored sites.

Existing approaches. Earlier approaches in air quality estimation employ statistical and machine learning techniques, such as Inverse Distance Weighting (IDW), Ordinary Kriging (OK), and Ordinary Cokriging (OCK) [9]. However, such non-learning approaches frequently need domain expertise to create particular settings for best performance. Moreover, nonlinear interactions between air indices and other variables, e.g. distance and meteorological factors, cannot be modeled. Recent works attempt to leverage the deep learning architecture to learn more complex spatio-temporal characteristics. Guo et al. proposed KIDW-TCGRU [5], which employs K-nearest Inverted Distance Weighting (K-IDW) to generate interpolated feature embedding prior to passing to the Time Distributed Convolutional Gated Recurrent Unit (TCGRU) model to extract the spatial-temporal characteristics and estimate the air quality. On the other hand, Ma et al. [11] developed a method combining a Bidirectional Long-short-term-memory network (BLSTM) and Inverse Distance Weighting (IDW) to cover the areas without monitoring stations. Using auto-encoder and a fully-connected network, [14] proposes a deep learning architecture to perform both interpolation, prediction and feature analysis of fine-grained air quality in one unified model.

Problem statement. Despite the fact that the spatial air quality estimation problem has gained increasing attention in recent years, existing approaches have not achieved high accuracy due to the following challenges. First, due to the lack of historical data at the targeted site, it is impossible to model the data distribution at that location, hence diminishing the estimation accuracy. To this end, a popular strategy is based on the data collected from nearby stations and their distance from the target location. In general, the majority of available methodologies are founded on the hypothesis that the shorter the distance between two locations, the greater the correlation between their air quality. However, the relationship between distance and air quality is significantly more complex and dependent on other variables, such as time and climate.

Our contribution. This study offers a highly accurate spatial air quality estimation method using encoder-decoder deep neural networks to address the above-mentioned limitations. Specifically, we propose AGE, an attentive graph neural network in the encoder, to model the spatial relationship between air quality at different sites using an attentive neural network. In addition, at the decoder, we leverage Gated Recurrent Units to combine the historical embeddings extracted by the encoder to enhance estimation accuracy. The key contributions of our paper are summarized as follows.

- We address the challenge of air quality interpolation for unmonitored areas and provide a formulation of the mentioned problem. Then, we propose an encoder-decoder framework that resolves the previously stated issues.
- We design a new attentive unsupervised representation-learning encoder that is capable of extracting the spatial relations of monitoring areas. In addition, we use a contrastive training mechanism to train the encoder to improve its induction capacity and noise tolerance.
- To validate the performance of our model versus existing benchmarks, we conduct comprehensive experiments on real-world datasets. The empirical findings show that our method outperforms other contemporary baselines.

The remainder of the paper is structured as follows. We formulate the targeted problem, our design principle, and the framework overview in Section 2. In Section 3, the proposed self-supervised graph representation learning strategy is discussed in detail, followed by the implementation of the temporal prediction layer in Section 4. We evaluate our proposed solution and existing approaches in Section 5. Finally, we conclude the paper in Section 6.

2 PROBLEM FORMULATION AND OUR APPROACH

This section begins with a formulation of air quality interpolation problem, followed by a discussion of the design principle and an overview of our proposed approach.

2.1 Problem Formulation

Before going into the details, we start with some definitions.

Multivariate air quality data. Air quality interpolation in an unmonitored area based on data from the nearby monitoring stations. At the timestamp t, each monitor station S_i records x_i^t of the target air quality indicator $PM_{2.5}$ $x_{i\_0}^t$; n air quality related features $\{x_{i\_1}^t, \dots x_{i\_n}^t\}$; and m meteorology data (i.e. temperature, evaporation, wind speed, precipitation). Thus, the multivariate historical data at station i has the form $x_i = \{x_i^0 \dots x_i^{t-1}, x_i^t, \dots x_i^T\}$, with T being the current timestamp.

Monitor station grid. A monitor station grid $\mathbb{S}$ is formed by monitor stations located in various locations. Each grid monitor station $S_i \in \mathbb{S}$ is assigned a coordinate $d_i = (\varphi_i, \lambda_i)$, which is its latitude and longitude. The distance function $D(S_i, S_j)$ is used to calculate the distance between any two stations in the grid $S_i, S_j \in \mathbb{S}$. As the distance function, we use the Haversine function [19], which is commonly used to calculate sphere distance. We constrain that there is a maximum distance threshold D^* such that there exists a S_m satisfies $D(S_m, S_n) < D^*$ for each station S_n. In real-world settings, D^* rarely exceeds 200km [13].

Problem formulation. Given a monitor station grid $\mathbb{S} = S_1, \dots, S_k$, historical multivariate data $X = \{x_1, \dots, x_k\}$, and a target location S_x satisfying the distance constraint $D(S_x, S_k) < D^*$, the problem is to estimate the current air quality indicator x_0^T at S_x at the current time step T. Assuming that we have detailed coordination of the target location as well as neighboring stations $C = \{C_1, \dots, C_k, C_x\}$. Then, at the current time step T, our prediction model will return the estimated target air quality value at the arbitrary location S_x, as described by the following formula:

$$O_x^T = P(X, C), \tag{1}$$

where P denotes prediction model.

2.2 Design principle

We believe that a framework addressing the aforementioned challenges should tackle the following issues:

- **C1: Spatio-temporal dependency:** Fine-grained air quality is temporally and spatially dependent, which means that current air quality values are frequently related to historical data as well as air quality values at neighboring locations. The challenge here is to create an architecture capable of capturing both of these pieces of information at the same time.

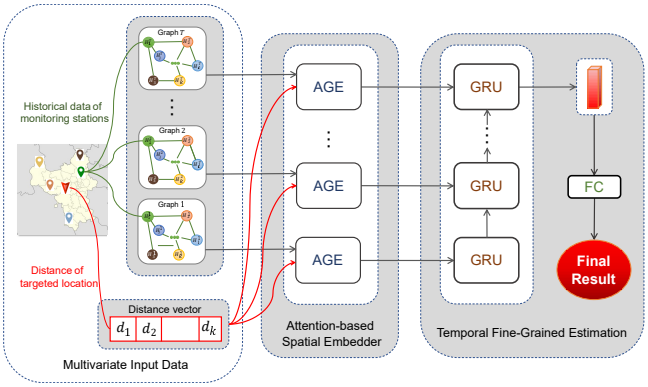


Figure 1: Overview architecture of our proposed framework

- **C2: Interpolation capability:** The absence of historical air quality data is a considerable obstacle to the estimation of air quality at any specified unknown location. This necessitates the framework's ability to model the correlation between locations based on available stations and generalize to arbitrage places. Existing interpolation techniques, such as Inverse Distance Weighting (IDW), may not fully model the non-Euclidean characteristics of the input data.

2.3 Framework Overview

With the problem **C1**, we first propose an attention-based spatio-graph neural network to capture the spatial information from the target node's nearby neighborhoods. Moreover, this component is capable of automatically learning the neighbor's importance by utilizing the attention mechanisms. To address **C2**, we employ the Gated Recurrent Network (GRU), which is well-known for its ability to handle sequence data effectively, so as to further enhance the performance of the predicting model.

To implement the functions discussed above, we design a framework using the unsupervised training paradigm, as illustrated in Figure 1. The encoder (spatial embedder) is specifically designed to take as input data a set of station monitoring networks from the current time-step G_T, where $G_T = (H_T, A)$, where H_T is the air quality and meteorology information network at the current time-step T, and A is the adjacency matrix calculated using equation 2. After that, the encoder is trained unsupervised using binary cross entropy loss to produce an output feature vector in low-embedding space. Section 3 goes over the specifics of this procedure.

Then, the trained encoder is applied multiple times to generate the historical embeddings for the target unknown station at each timestep. Those historical representations are fed through the decoder which consists of GRU and a fully connected layer to estimate the current air quality value at the unmonitored location. The decoder is described thoroughly in Section 4.

3 ATTENTION-BASED SPATIAL GRAPH REPRESENTATION LEARNING

This section begins with a description of the spatial-input graph's formation. Afterward, the attention-based unsupervised graph representation learning module is established, which aims to capture the spatial aspects of the input data. Finally, we discuss the method of learning unsupervised graph embeddings using the contrastive learning paradigm.

3.1 Generation of the spatio-network graph

Given the monitoring station data, at each given timestamp i-th, we generate a representative graph G_i. Each node in those graphs represents a station, and connections between nodes reflect the neighbor relationship between two respective stations. Denoted by n the number of monitoring stations and k the total number of air quality and meteorological features combined. The i-th timestamp graph is denoted as $G_i = \{\mathbf{X}_i, A\}$, where $\mathbf{X}_i \in R^{n*k}$ represents the information from all monitoring stations and $A \in R^{n*n}$ denotes the adjacency matrix. We construct the adjacency matrix following the below equation:

$$A_{ij} = \begin{cases} 0 & \text{if } distance(i,j) > threshold_{distance} \\ 1 & \text{if } distance(i,j) \leq threshold_{distance}, \end{cases} \quad (2)$$

where $distance(i, j)$ is obtained by applying the Haversine distance formula between the i-th station and the j-th station. $threshold_{distance}$ is a threshold that identifies the neighborhood of a node. Specifically, if the distance between two nodes exceeds the $threshold_{distance}$, they are not immediate neighbors and vice versa.

For the node features $\mathbf{X}_i$, we use the air quality related features such as $PM_{2.5}$, PM_{10}, NO_2, SO_2, O_3, and meteorology information, i.e. temperature, precipitation, wind speed and evaporation.

3.2 Attention-based inductive embedding generation

In this section, we will present the graph attention network layer and inductive graph framework that have the capability to learn the representation of unseen nodes. We first describe the graph attention layer which adaptively selects k immediate nearest neighbors and produces the node embedding based on the importance scores of those neighbors. Then, we elaborate on the inductive graph embedder for generating the embeddings of unseen nodes, which is followed by detailed algorithms for the training and inference process.

Graph Attention Layer. Since various stations have different impacts on the considering location, it is necessary to investigate and assess the importance of each station. Other approaches (e.g. KIDW-TCGRU [5], IDW-BILSTM [11]) rely only on distance data, which does not represent non-Euclidean relations. These coefficients can be learned automatically by utilizing the attention method. Therefore, we propose employing the node representation learning method with the attention mechanism according to the AffinityNet model. Introduced in the Affinitynet model by T. Ma et al. [10], kNN attention layer encourages adjacent nodes in a graph to have similar representations while ensuring that representations of disparate nodes are highly separate. This layer also provides the attention score, which indicates the significance of the neighborhood across the considering node. kNN attention layer calculates the transformed feature representation of a graph node using attention-based polling:

$$\mathbf{h}_i' = f\left(\sum_{j \in \mathcal{N}(i)} a\left(\mathbf{h}_i, \mathbf{h}_j\right) \cdot \mathbf{h}_j\right) \quad (3)$$

where $\mathbf{h}_i$ is the input feature representation for node i and $\mathbf{h}'_i$ is its transformed feature representation. $\mathcal{N}(i)$ denotes the immediate neighborhood of node j. $a\left(\mathbf{h}_i, \mathbf{h}_j\right)$ is the normalized attention from object i to object j, which is calculated by:

$$a\left(\mathbf{h}_i, \mathbf{h}_j\right) = \frac{e^{\alpha_{ij}}}{\sum_{j \in \mathcal{N}(i)} e^{\alpha_{ij}}} \tag{4}$$

where α_{ij} is the weight computed according to an attention kernel. In this work, the inner product (Vaswani et al. [16]) is chosen as the kernel:

$$\alpha_{ij} = \mathbf{h}_i \cdot \mathbf{h}_j \tag{5}$$

Spatial Embedder. While the proposed graph attention layer aims to learn the node embeddings, it has limited capability in realizing the embeddings of unseen nodes. Therefore, we propose a different graph learning mechanism, which is inspired by the GraphSAGE [6] algorithm, to produce an inductive embedder that can generalize rapidly for new nodes. Specifically, we propose our AGE (**A**ttention-based **G**raph **E**mbedder) to capture the spatial relation between the input grid stations.

Given the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of monitoring stations and $\mathcal{E}$ is the set of connections between these locations. Each station v in $\mathcal{V}$ has feature $\mathbf{x}_v$ which is provided as input of the AGE algorithm. $\mathcal{N}(v)$ denotes the immediate neighborhood of node v. The number of graph convolutional layers K - the number of hops via which node information is aggregated during each iteration, is a critical hyperparameter of this algorithm. The number of graph layers K is a critical hyperparameter of this algorithm. It defines the depth up to which neighborhood information can be aggregated to the target location. Another key component of AGE is aggregator architecture, in which we use the kNN attention polling model. We denote $\mathbf{h}_v^k$ as the embedding of node v in depth k. Input features can be assigned as $\mathbf{h}_v^0$. Each depth layer k has a distinctive attention aggregator, which is denoted by KNN-ATT$_k$. Each node $v \in \mathcal{V}$ aggregates the representations of all the nodes in its immediate neighborhood $\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\}$ into a single vector $\mathbf{h}_{\mathcal{N}(v)}^k$:

$$\mathbf{h}_{\mathcal{N}(v)}^k \leftarrow \text{KNN-ATT}_k\left(\left\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\right\}\right) \tag{6}$$

Then, a node's previous-depth representation $\mathbf{h}_v^{k-1}$ is concatenated with its aggregated neighborhood vector of the current depth $\mathbf{h}_{\mathcal{N}(v)}^k$, whose result is passed through a fully connected layer with ReLU activation function. This process can be described as follows:

$$\mathbf{h}_v^k \leftarrow \sigma\left(\mathbf{W}^k \cdot \text{CONCAT}\left(\mathbf{h}_v^{k-1}, \mathbf{h}_{\mathcal{N}(v)}^k\right)\right) \tag{7}$$

After K iterations, each node representation is aggregated from its immediate neighborhood up to its K-hop neighbors. Moreover, the aggregation attention weight is automatically learned when using the KNN-ATT function.

3.3 Contrastive self-supervised training

To effectively train the embeddings, we also devise a self-supervised adversarial learning process (see Figure 2). This component consists of two major parts: the *graph corruptor* and the *discriminator*.

Graph corruptor. By injecting the small perturbations into the original graph, we produce additional training scenarios. Diverse contexts promote the generalizability of the output embeddings in subsequent adversarial training since the difference between several perspectives of the same original graph gives additional supervisory signals. To achieve this, the graph corruptor is designed to enhance the original graph's structure and attributes.

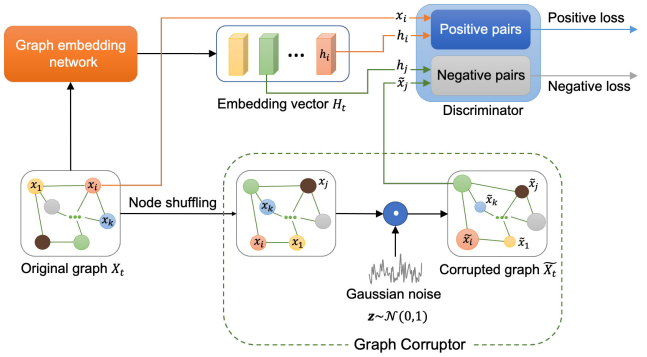


Figure 2: Unsupervised training process

For the structural perturbation, we utilize the row-by-row reshuffling procedure as described in [12]. This procedure entails a change in the topological structure of the network due to the random exchange of node feature values. Then, each node's feature is corrupted by adding Gaussian noise to its property. More specifically, we first sample a random vector $\tilde{m} \in \{0,1\}^F$, where each dimension is drawn individually from a Gaussian distribution. Afterward, the corrupted node's attributes $\tilde{\mathbf{X}}$ is computed by:

$$\tilde{\mathbf{X}} = [x_1 \circ \tilde{m}; x_2 \circ \tilde{m}; \cdots; x_N \circ \tilde{m}]^\top \tag{8}$$

where $\circ$ is the concatenation operator.

Discriminator. The discriminator is trained using the principle of mutual information (MI) maximization, where the mutual information between each node embedding and its appropriate node feature is maximized, and minimized otherwise. Particularly, given the embeddings and the raw features $\mathbf{X}_t$, and the corrupted features $\tilde{\mathbf{X}}_t$ retrieved by applying the graph corruptor on $\mathbf{X}_t$, we denote a pair of positive sample as (h_v^t, x_v^t), and pair of negative sample as $(h_v^t, \tilde{x}_v^t)$, where h_v^t is the spatial embedding of node v at time-step t using the encoder ϵ, x_v^t, and $\tilde{x}_v^t$ are retrieved from $\mathbf{X}_t$, and $\tilde{\mathbf{X}}_t$, respectively.

Then, the discriminator $\mathcal{D}$ produces the probability of (x_i, h_i) - a positive pair by applying a bilinear scoring function:

$$\mathcal{D}\left(\vec{x}_i, \vec{h}_i\right) = \sigma\left(\vec{h}_i^T \mathbf{W} \vec{x}_i\right) \tag{9}$$

where h_i and x_i are the embedding and the raw features of the same node, respectively. W is the learnable matrix and σ is the logistic sigmoid activation function. Next, we combine and optimize the contrastive loss $\mathcal{L}_{ssl}$, which is described as follows:

$$\mathcal{L}_{ssl} = \frac{1}{2N}\left(\sum_{i=1}^{N} \mathbb{E}_{(\mathrm{X},\mathrm{A})}\left[\log \mathcal{D}\left(x_i, h_i\right)\right] + \sum_{j=1}^{N} \mathbb{E}_{(\tilde{\mathrm{X}},\tilde{\mathrm{A}})}\left[\log\left(1 - \mathcal{D}\left(\tilde{x}_j, h_j\right)\right)\right],\right. \tag{10}$$

where N is the number of the positive and negative samples. The encoder is then trained independently using this loss function, and the Adam optimizer [8] is used to update the parameter weights after each epoch.

The result of this training process, which is a set of aggregator functions capable of generalizing for unseen nodes with input features, will be applied differently in the inference phase. Given that the target location has no previous air quality indices or meteorological data, it cannot be directly fed through AGE, which requires both

Algorithm 1: Training process

Input: Graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$; historical input features $\left\{\mathbf{x}_v^i, \forall v \in \mathcal{V}, \forall \text{ time steps } i = 1, \ldots, T\right\}$; depth K; weight matrices $\mathbf{W}^k, \forall k \in \{1, \ldots, K\}$; non-linearity σ; differentiable aggregator functions KNN-ATT$_k, \forall k \in \{1, \ldots, K\}$; neighborhood function $\mathcal{N} : v \rightarrow 2^{\mathcal{V}}$; corruptor C; neighborhood $\mathcal{N}(z)$ of target node z

Output: Current air quality value at target location $\mathbf{O}_z^T$

$\mathbf{h}_v^0 \leftarrow \mathbf{x}_v^T, \forall v \in \mathcal{V}$
repeat
  for $k = 1, \ldots, K$ **do**
    for $v \in \mathcal{V}$ **do**
      $\mathbf{h}_{\mathcal{N}(v)}^k \leftarrow \text{KNN-ATT}_k\left(\left\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\right\}\right)$
      $\mathbf{h}_v^k \leftarrow \sigma\left(\mathbf{W}^k \cdot \text{CONCAT}\left(\mathbf{h}_v^{k-1}, \mathbf{h}_{\mathcal{N}(v)}^k\right)\right)$
    end
  end
  $H = \left\{\mathbf{h}_v^K, v \in \mathcal{V}\right\}$
  $\tilde{H} = C\,(H)$
  Compute the probability of each sample pair being positive following Equation 9
  Compute loss $\mathcal{L}_{ssl}$ following Equation 10
  Update the encoder parameters
until *reach convergence*;
Frozen the parameter of encoder network
for $i = 1, \ldots, T$ **do**
  $\mathbf{h}_v^0 \leftarrow \mathbf{x}_v^i, \forall v \in \mathcal{V}$
  for $k = 1, \ldots, K$ **do**
    for $v \in \mathcal{V}$ **do**
      $\mathbf{h}_{\mathcal{N}(v)}^k \leftarrow \text{KNN-ATT}_k\left(\left\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\right\}\right)$
      $\mathbf{h}_v^k \leftarrow \sigma\left(\mathbf{W}^k \cdot \text{CONCAT}\left(\mathbf{h}_v^{k-1}, \mathbf{h}_{\mathcal{N}(v)}^k\right)\right)$
    end
    $\mathbf{h}_{\mathcal{N}(z)}^k \leftarrow \text{KNN-ATT}_k\left(\left\{\mathbf{h}_t^{k-1}, \forall t \in \mathcal{N}(z)\right\}\right)$
    $\mathbf{h}_z^k \leftarrow \sigma\left(\mathbf{W}^k \cdot \text{CONCAT}\left(\mathbf{h}_z^{k-1}, \mathbf{h}_{\mathcal{N}(z)}^k\right)\right)$
  end
  $\mathbf{h}_z^i \leftarrow \mathbf{h}_z^K$
end
repeat
  for $t = 1, \ldots, T$ **do**
    Compute $\bar{h}_t$ following Equation 11
  end
  Compute $\mathbf{O}_z^T$ following Equation 12
  Compute the *MSE* loss following the Equation 13
  Update the decoder parameters
until *reach convergence*;

air quality and climatic data as input. Instead, its representation will be passively interpolated from other monitoring stations using aggregator functions. In other words, the embedding of the target node in each iteration will neither affect the representations of its neighborhood nor update the parameters of aggregator functions.

The whole training and inference process of AGE can be expressed from lines 1 to 27 of Algorithm 1.

4 NEURAL-BASED TEMPORAL ESTIMATION

4.1 Temporal air quality prediction

In this section, we elaborate on the architecture of the decoder, which consists of two main components: *Temporal extractor* and *Predictor*.

Temporal extractor. To enhance the performance of the predicting model, we leverage not only the current spatial information retrieved from the encoder but also embeddings of the past timesteps. These historical embeddings can be generated using the AGE architecture with input being the graphs at previous timestamps. Moreover, we need to select a suitable architecture to better capture the temporal information across timesteps. In our proposed method, we choose the GRU network [3] due to two reasons: (i) it overcomes the gradient issues of vanilla recurrent neural network (RNN) (i.e. gradient vanishing, gradient explosion) by employing the gated mechanism; and (ii) its structure is relatively simple compared to that of LSTM, another common RNN version, leading to a smaller total training time and fewer parameters.

The graph embeddings and GRU-based extractor are combined to learn the spatio-temporal dynamic of input data:

$$\begin{aligned} r_t &= \sigma\left(W_r AGE(\mathbf{X}_t, A) + U_r \bar{h}_{(t-1)} + b_r\right), \\ z_t &= \sigma\left(W_z AGE(\mathbf{X}_t, A) + U_z \bar{h}_{(t-1)} + b_z\right), \\ c_t &= \tanh\left(W_n AGE(\mathbf{X}_i, A) + \bar{h}_{(t-1)}\,(R_t \odot W_n) + b_n\right), \\ \bar{h}_t &= (1 - z_t) \odot c_t + z_t \odot \bar{h}_{(t-1)}, \end{aligned} \tag{11}$$

where the outputs at time step t of the graph embedding generations $AGE(\mathbf{X}_t, A) = h_z^t$ and the hidden feature at the prior timestep $\bar{h}_{(t-1)}$ are fed into the temporal extractor. W and U denote the weight matrices for each control gate and the b terms are bias vectors, $\odot$ denotes the Hadamard product, σ denotes the activation function, and r_t, z_t, c_t, and $\bar{h}_t$ are the reset gate, update gate, candidate hidden state, and hidden state, respectively.

Predictor. Given $\bar{h}_T$ is the feature embedding of the *temporal extractor*, we then calculate the current fine-grained air quality level at the target location $\mathbf{O}_z^T$. Specifically, $\bar{h}_T$ is forwarded to the fully-connected layer, which is mathematically described as follows:

$$\mathbf{O}_z^T = \sigma(FC(\bar{h}_T)) \tag{12}$$

where σ is the ReLU activation function.

We then leverage the MSE (Mean Square Error) loss function to train the decoder. The formula of MSE loss is as follows:

$$MSE = \frac{1}{N} \sum_{t=1}^{N} (y_t - \tilde{y}_t)^2 \tag{13}$$

where y_t and $\tilde{y_t}$ are the predicted fine-grained air quality value at the target point and the ground truth, respectively, N is the number of observations in the input data batch.

The overall training process of our proposed model is summarized in Algorithm 1.

Algorithm 2: Testing process

Input: Graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$; historical input features $\left\{\mathbf{x}_v^i, \forall v \in \mathcal{V}, \forall \text{ time steps } i = 1, \ldots, T\right\}$; depth K; weight matrices $\mathbf{W}^k, \forall k \in \{1, \ldots, K\}$; non-linearity σ; trained aggregator functions KNN-ATT$_k, \forall k \in \{1, \ldots, K\}$; neighborhood function $\mathcal{N} : v \rightarrow 2^{\mathcal{V}}$; trained decoder parameters; neighborhood $\mathcal{N}(z)$ of target node z

Output: Current air quality value at target location $\mathbf{O}_z^T$

$\mathbf{h}_v^0 \leftarrow \mathbf{x}_v^T, \forall v \in \mathcal{V}$
for $i = 1, \ldots, T$ **do**
  $\mathbf{h}_v^0 \leftarrow \mathbf{x}_v^i, \forall v \in \mathcal{V}$
  for $k = 1, \ldots, K$ **do**
    for $v \in \mathcal{V}$ **do**
      $\mathbf{h}_{\mathcal{N}(v)}^k \leftarrow \text{KNN-ATT}_k\left(\left\{\mathbf{h}_u^{k-1}, \forall u \in \mathcal{N}(v)\right\}\right)$
      $\mathbf{h}_v^k \leftarrow \sigma\left(\mathbf{W}^k \cdot \text{CONCAT}\left(\mathbf{h}_v^{k-1}, \mathbf{h}_{\mathcal{N}(v)}^k\right)\right)$
    end
    $\mathbf{h}_{\mathcal{N}(z)}^k \leftarrow \text{KNN-ATT}_k\left(\left\{\mathbf{h}_t^{k-1}, \forall t \in \mathcal{N}(z)\right\}\right)$
    $\mathbf{h}_z^k \leftarrow \sigma\left(\mathbf{W}^k \cdot \text{CONCAT}\left(\mathbf{h}_z^{k-1}, \mathbf{h}_{\mathcal{N}(z)}^k\right)\right)$
  end
  $\mathbf{h}_z^i \leftarrow \mathbf{h}_z^K$
end
for $t = 1, \ldots, T$ **do**
  Compute $\bar{h}_t$ following Equation 11
end
Compute $\mathbf{O}_z^T$ following Equation 12

4.2 Air quality interpolation for unmeasured locations

Given the trained model and input data from the monitoring stations, we demonstrate the interpolation procedure for estimating the air quality in an unsupervised region that has been selected. Algorithm 2 describes the comprehensive inference procedure of our architecture.

5 PERFORMANCE EVALUATION

In this section, experiments are conducted to address the following research questions:

- **Q1** Does our proposed model outperform the benchmark methods?
- **Q2** How crucial is each input feature for estimating air quality?

In the subsequent sections, we discuss the dataset utilized throughout the experiment in 5.1 and the experimental settings in 5.2. Then, to address the specified issues, we undertake our empirical evaluations, including an end-to-end comparison in 5.3 and a feature importance in 5.4.

5.1 Datasets

Beijing Dataset The Beijing dataset [18] contains air quality and meteorological data from 35 stations in Beijing in 2018. This dataset has a total area of 16,441 km2, the majority of which consists of metropolitan regions with industrial zones and intense transportation networks; hence, high air quality indices are typically observed. This dataset comprises six types of hourly recorded pollutants, including $PM_{2.5}$, PM_{10}, NO_2, CO, SO_2, and O_3. In addition, meteorological characteristics such as temperature, pressure, precipitation and wind speed are recorded.

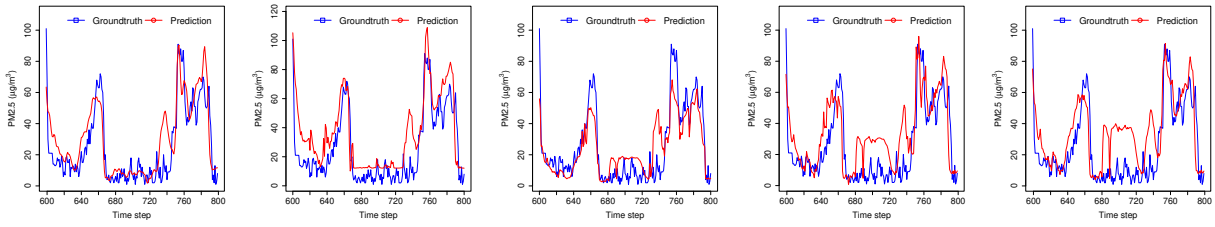
UK Dataset Reani et al. [15] presented in 2021 a dataset of daily UK meteorology, air quality, and pollen readings from 2016 to 2019, spanning four consecutive years. This dataset covers a total area of 242,295 km^2 and contains a variety of topographic features, including harsh, underdeveloped hills, low ranges, and rolling plains. The authors gathered air temperature ($^\circ C$), humidity (rH), precipitation (mm) and wind speed (m/s) for the meteorological data set. Otherwise, the air pollutant dataset includes O_3, NO_2, SO_2, PM_{10} and $PM_{2.5}$. The dataset contains information from 141 air quality monitoring sites around the United Kingdom. If the distance between two neighboring stations is less than 200 kilometers, only 30 stations matching this criterion are used.

5.2 Setting

Metrics. In this study, we analyze the performance of models using five statistical metrics: root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), median absolute percentage error (MDAPE), and $R2$ Score.

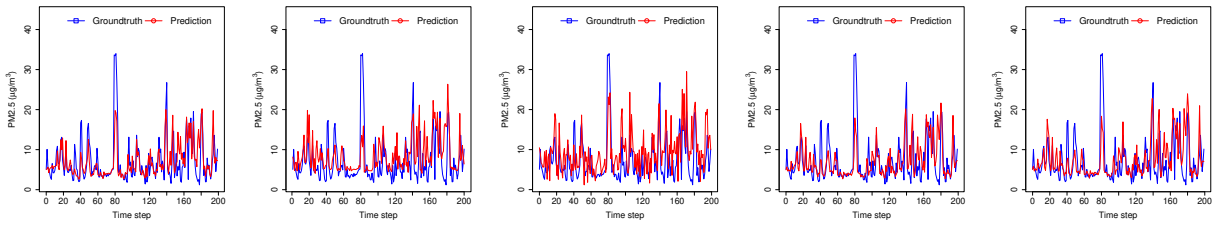
Benchmarks. To evaluate the performance of our proposed model, we conduct a fair comparison with the deep learning-based algorithms KIDW-TCGRU, BiLSTM-IDW, AttPolling FCNN, and FCNN.

- **BiLSTM-IDW:** was introduced in 2019 by Ma et al [11]. It is a two-phased model that uses BiLSTM to train the output feature embedding and Invert Distance Weight (IDW) to aggregate the feature embedding using a linear distance function. The aggregated output feature is then sent to a prediction layer, which generates the final interpolation value.
- **KIDW-TCGRU:** Guo et al proposed KIDW-TCGRU [5] in 2020 . This method combines the inverse distance weighting KNN (IDW-KNN) with the TCGRU model. The objective of the IDW-KNN approach is to identify the closest stations for interpolation. The TCGRU model, which is comprised of a time-distributed convolutional neural network (TCNN) and a gated recurrent network (GRU), helps the model learn spatial and temporal properties.
- **AttPolling FCNN:** was introduced by Colchado et al [4]. This technique introduces a model for deep learning based on an attention mechanism that learns the effect score of neighboring nodes irrespective of distance information. Then, the prediction layer, which is composed of numerous Fully-connected layers, integrates the weighted feature vectors from neighboring nodes to compute the $PM_{2.5}$ concentration value at the target site.
- **FCNN:** This technique is a simplified version of the Attention-Polling FCNN technique. However, rather than automatically locating neighboring nodes, these stations are decided based on distance information. This method then takes the meteorological information and $PM_{2.5}$ value from neighboring nodes as input for the prediction layer, which comprises of many fully-connected layers, in order to calculate the interpolated value.



(a) AGE (b) IDW-BiLSTM (c) KIDW-TCGRU (d) AttPolling FCNN (e) FCNN

Figure 3: Prediction results on Beijing dataset



(a) AGE (b) IDW-BiLSTM (c) KIDW-TCGRU (d) AttPolling FCNN (e) FCNN

Figure 4: Prediction results on UK dataset

Table 1: Average accuracy of air quality estimation methods

Dataset	Model	MAE	RMSE	MAPE	MDAPE	R2
Beijing	AGE	**10.25**	**14.92**	**0.60**	**0.235**	**0.89**
	BiLSTM-IDW	13.25	18.28	0.61	0.322	0.85
	KIDW-TCGRU	16.28	20.38	0.78	0.432	0.76
	AttPolling FCNN	11.15	15.03	0.63	0.252	0.87
	FCNN	11.70	15.36	0.67	0.260	0.86
UK	AGE	**1.81**	**2.74**	**0.30**	**0.195**	**0.54**
	BiLSTM-IDW	2.59	3.6	0.39	0.390	0.42
	KIDW-TCGRU	2.85	4.18	0.52	0.520	0.43
	AttPolling FCNN	2.32	3.35	0.37	0.257	0.48
	FCNN	2.33	3.58	0.38	0.297	0.45

5.3 End-to-end comparison

In this part, we compare the precision of our method (AGE) to the baseline methods so as to answer the question **RQ1**. Overall, the performance of our proposed model surpasses the performance of previous baseline models in both datasets. In terms of MAE error, our suggested model improves from 21.83% to 36.37% for the UK dataset, whereas this number spans from 8.07% to 37.04% for the Beijing dataset. For additional measurements, the experimental results of the proposed method remain superior to those of contemporary methods. The table 1 demonstrates the detailed performances of proposed approaches and baseline models across both datasets.

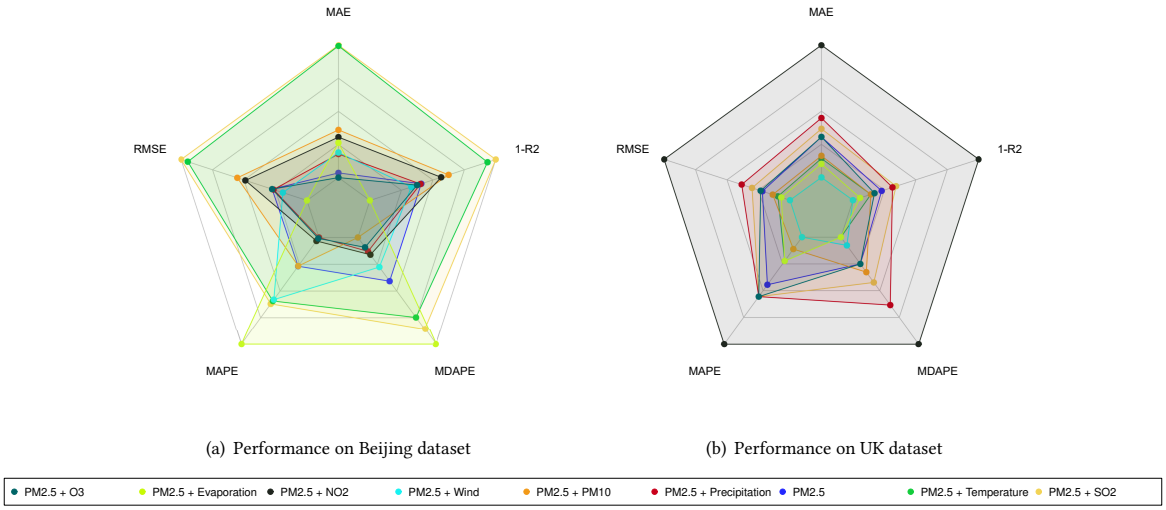
In each dataset, we show the expected air quality value of the proposed model and baseline methodology for a representative station, which are shown in Figure 3 and 4. In both figures, the proposed method forecasts low points substantially more accurately than prior methods. However, owing to the nature of the technique, which strongly relies on the $PM_{2.5}$ values of neighboring stations, the anticipated value trend remains consistent with the trends of neighboring stations. An illustration of this proposed method's problem is the surge in the value of predicted data on the Beijing dataset between the 720^{th} timestep and the 750^{th} timestep.

5.4 Feature importance analysis

In order to answer the question **RQ2**, we perform a series of experiments, the first of which uses solely $PM_{2.5}$ and serves as the baseline. Then, we iteratively add one of the remaining features and record the performance of the model. Combining data from satellite stations [7] with nearby observation stations, we collected some more potential features, from which we chose evaporation to perform experiments on feature importance. Notably, we pick a subset of typical characteristics from both datasets to conduct these experiments, including: $PM_{2.5}$, NO_2, PM_{10}, SO_2, O_3, precipitation, temperature, wind speed and evaporation. The results of each experiment are depicted in Figure 5. Accordingly, we may infer that the most important characteristics for interpolation of $PM_{2.5}$'s concentration are: $PM_{2.5}$, O_3, SO_2, PM_{10}, evaporation, evaporation and wind speed, while the contribution of NO_2 and temperature and are less significant. Because of the temporal correlation, the previous $PM_{2.5}$'s concentration is most equivalent to the upcoming $PM_{2.5}$'s value. PM_{10} is also highly relevant to $PM_{2.5}$ [14]. Evaporation, which is a meteorological attribute indicates the amount of evaporated water in the air, shows a strong relation with $PM_{2.5}$'s concentration [2]. Furthermore, wind speed have strong correlation with the $PM_{2.5}$'s concentration [1], which results in the significant contribution of this feature to the interpolation process.

6 CONCLUSION

This research established a novel framework for graph self-supervised representation learning on spatio-temporal data and resolved the challenge of estimating fine-grained air quality at unmonitored



(a) Performance on Beijing dataset (b) Performance on UK dataset

Figure 5: Impacts of different features on the PM2.5 prediction accuracy.

locations. Specifically, we propose a novel attentive interpolation learning method that is capable of modeling the distance and non-Euclidean spatial relations between areas. This approach utilizes the contrastive learning paradigm to embed characteristics of the input graph into low-dimensional vectors. Then, with the intention of predicting air quality indicator $PM_{2.5}$, we apply the GRU model to the historical embedding vectors for unmonitored areas in order to get a final representation and prediction of the current timestamp. Our approach estimates the air quality value at unknown regions by combining the target historical representations embedded from surrounding monitoring stations. The experimental results indicate that the proposed model provides more reliable predictions than state-of-the-art models. Specifically, in terms of the MAE metric, our proposed model outperforms other baselines by 8.07% to 37.04%.

ACKNOWLEDGMENTS

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.01-2019.302.

REFERENCES

[1] Yuanchen Chen, Lu Zang, Wei Du, Da Xu, Guofeng Shen, Quan Zhang, Qiaoli Zou, Jinyuan Chen, Meirong Zhao, and Defei Yao. 2018. Ambient air pollution of particles and gas pollutants, and the predicted health risks from long-term exposure to PM2. 5 in Zhejiang province, China. *Environmental Science and Pollution Research* 25, 24 (2018), 23833–23844.

[2] Ziyue Chen et al. 2018. Understanding meteorological influences on PM 2.5 concentrations across China: a temporal and spatial perspective. *Atmospheric Chemistry and Physics* 18, 8 (2018), 5343–5358.

[3] Junyoung Chung et al. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014).

[4] Luis E Colchado et al. 2021. A Neural Network Architecture with an Attention-based Layer for Spatial Prediction of Fine Particulate Matter. In *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 1–10.

[5] Canyang Guo et al. 2020. An unsupervised PM2. 5 estimation method with different spatio-temporal resolutions based on KIDW-TCGRU. *IEEE Access* 8 (2020), 190263–190276.

[6] Will Hamilton et al. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[7] H. Hersbach et al. 2018. ERA5 hourly data on single levels from 1959 to present.

[8] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[9] Jin Li et al. 2011. A review of comparative studies of spatial interpolation methods in environmental sciences: Performance and impact factors. *Ecological Informatics* 6, 3-4 (2011), 228–241.

[10] Tianle Ma and Aidong Zhang. 2019. Affinitynet: semi-supervised few-shot learning for disease type prediction. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 1069–1076.

[11] Jun Ma et al. 2019. Spatiotemporal prediction of PM2. 5 concentrations at different time granularities using IDW-BLSTM. *IEEE Access* 7 (2019), 107897–107907.

[12] Felix L Opolka et al. 2019. Spatio-temporal deep graph infomax. *arXiv preprint arXiv:1904.06316* (2019).

[13] Yanlin Qi et al. 2019. A hybrid model for spatiotemporal forecasting of PM2. 5 based on graph convolutional neural network and long short-term memory. *Science of the Total Environment* 664 (2019), 1–10.

[14] Zhongang Qi et al. 2018. Deep air learning: Interpolation, prediction, and feature analysis of fine-grained air quality. *IEEE Transactions on Knowledge and Data Engineering* 30, 12 (2018), 2285–2297.

[15] Manuele Reani et al. 2022. UK daily meteorology, air quality, and pollen measurements for 2016–2019, with estimates for missing data. 9, 1 (2022), 1–12.

[16] Ashish Vaswani et al. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[17] W. H. O. (WHO). 2016. Ambient air pollution: A global assessment of exposure and burden of disease.

[18] Hongwei Wang. 2019. Air pollution and meteorological data in Beijing 2017-2018.

[19] Edy Winarno et al. 2017. Location based service for presence system using haversine method. In *2017 international conference on innovative and creative information technology (ICITech)*. IEEE, 1–4.