Self-consistent Deep Geometric Learning for Heterogeneous Multi-source Spatial Point Data Prediction

Dazhou Yu
dyu62@emory.edu
Emory University
Atlanta, GA, United States

Xiaoyun Gong
kristinagxy51@gmail.com
Emory University
Atlanta, GA, United States

Yun Li
yli230@emory.edu
Emory University
Atlanta, GA, United States

Meikang Qiu
qiumeikang@yahoo.com
Augusta University
Augusta, GA, United States

Liang Zhao
liang.zhao@emory.edu
Emory University
Atlanta, GA, United States

Abstract

Multi-source spatial point data prediction is crucial in fields like environmental monitoring and natural resource management, where integrating data from various sensors is the key to achieving a holistic environmental understanding. Existing models in this area often fall short due to their domain-specific nature and lack a strategy for integrating information from various sources in the absence of ground truth labels. Key challenges include evaluating the quality of different data sources and modeling spatial relationships among them effectively. Addressing these issues, we introduce an innovative multi-source spatial point data prediction framework that adeptly aligns information from varied sources without relying on ground truth labels. A unique aspect of our method is the 'fidelity score,' a quantitative measure for evaluating the reliability of each data source. Furthermore, we develop a geo-location-aware graph neural network tailored to accurately depict spatial relationships between data points. Our framework has been rigorously tested on three real-world datasets and one synthetic dataset. The results consistently demonstrate its superior performance over existing state-of-the-art methods.

CCS Concepts

• **Computing methodologies** → **Machine learning**.

Keywords

multi-source; spatial data; point data; data fusion; graph neural networks

ACM Reference Format:
Dazhou Yu, Xiaoyun Gong, Yun Li, Meikang Qiu, and Liang Zhao. 2024. Self-consistent Deep Geometric Learning for Heterogeneous Multi-source Spatial Point Data Prediction . In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), August 25–29, 2024, Barcelona, Spain.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3637528.3671737

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
KDD '24, August 25–29, 2024, Barcelona, Spain
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0490-1/24/08
https://doi.org/10.1145/3637528.3671737

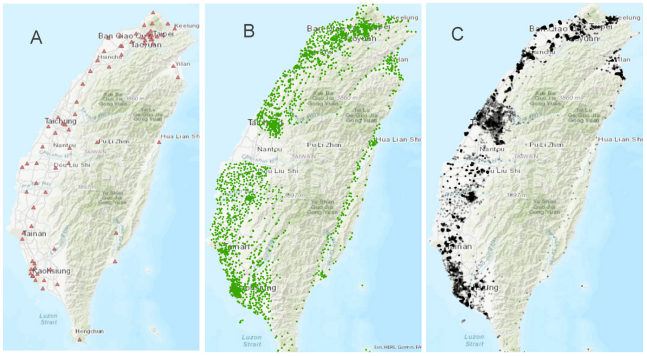

Figure 1: An example of multi-source spatial point prediction problem: Varying distribution of three data sources including (A) 74 AQMSs, (B) 3704 LASS AirBox sensors, and (C) 9701 EPA MicroStations.

1 Introduction

Spatial prediction [40, 42, 43] is essential in various domains such as environmental monitoring, natural resource management, and transportation planning [13, 15]. The process involves collecting various attributes and utilizing them to fit predictive models for target variables. PM 2.5 prediction is a popular application of spatial point prediction, where the data used for model training and prediction could be emission data or observations from fixed sites of air quality monitoring stations (AQMS). Although AQMSs offer high-quality data, their deployment and maintenance are costly, leading to insufficient coverage in extensive areas [17]. To compensate for these coverage gaps, a common practice is to deploy numerous low-cost miniature sensors [3, 4, 10, 24]. The cost of one AQMS is above one hundred times the cost of one microsensor. Thus, using the same budget, the deployment of microsensors might reach a density of observations 100 times higher than the deployment of regulatory AQMSs. Figure 1 from [4] shows distributions of three different types of data sources in Taiwan, including AQMSs, AirBox sensors maintained by Academia Sinica [3], and the Taiwan EPA's Air Quality MicroStations [1]. The quality of the low-cost sensors is not as good as AQMSs. Additionally, the range

of attributes collected varies between these sources, with certain pollutants like nonmethane hydrocarbons being exclusively monitored by AQMSs. To fully exploit each source's unique properties, a specialized approach to multi-source spatial data prediction is needed.

Developing a machine learning model that effectively caters to the intricacies of multi-source spatial point data prediction remains difficult, due to several open challenges: **1) Difficulty in effectively aligning information across data sources without ground truth.** Data from different sources may exhibit similar trends, yet they often have conflicting information. Each source, with its unique characteristics, could potentially complement another. In the absence of ground truth data, developing a robust and effective training strategy that can align these varied sources is crucial for maximizing their collective utility. **2) Difficulty in aligning information across various data source qualities.** The sensor network, comprised of sensors with varying qualities, poses a challenge in aligning information. It is essential to quantify the quality of different sources accurately to prevent the dilution of high-quality data with noise from lower-quality sources. Despite having some prior knowledge about the accuracy of certain sources, quantifying these quality levels and effectively utilizing the data from lower-quality sources remain challenging. **3) Difficulty in aligning information across different spatial locations.** The locations of samples are usually different across data sources. Real-world spatial points usually exhibit spatial autocorrelation. However, existing methods struggle to capture the complex spatial relationship patterns when samples come from different locations and data sources. An ideal method should account for the relative positions of all neighbors, adapting to the dynamic influences of the surrounding environment.

To address these challenges, we introduce a comprehensive framework for multi-source spatial point data prediction. This framework is designed with specific components to tackle each identified challenge: To address the first challenge, we propose a self-supervised strategy derived from maximizing the mutual information between the model estimation and the target variable across each data source. This approach aims to align data from all sources in a self-supervised manner, leveraging their collective strengths for enhanced prediction accuracy. To address the second challenge, we introduce the concept of a fidelity score which is derived from the likelihood that the target variable in a data source aligns with the ground truth. This parameter is unconstrained and designed to be learnable, allowing the model to adjust and quantify the quality of each data source dynamically using gradient descent algorithms. For the third challenge, we go beyond existing methodologies by developing a geo-location-aware multi-source graph neural network. This network is uniquely equipped with shared and distinct modules to process heterogeneous types of features, thereby effectively capturing the complex spatial relationships across different data sources. Our contributions are thus encapsulated as follows:

- We propose the Deep Multi-source Spatial Prediction (DMSP) framework that aligns diverse data sources in a self-supervised manner.
- We introduce the concept of fidelity score, a learnable parameter for quantifying the quality of each data source, enabling the effective utilization of mixed-quality data.
- We propose a geo-location-aware multi-source graph neural network designed to handle the complexities of spatial relationships and feature heterogeneity across different data sources.
- We conduct extensive experiments on three public real-world datasets and one synthetic dataset to validate the superiority of our method.

2 Related Work

In recent years, various methods have emerged for multi-source spatial point prediction, broadly categorized into three groups: 1) Traditional models. The traditional theory-based models are often based on in-depth domain knowledge, such as those used in air pollutant modeling [18, 29, 36]. They offer interpretability but face limitations due to computational demands and the quality of available data sources [35]. In complex scenarios like river network modeling [32], the multitude of influencing factors makes mechanistic characterization challenging. A critical drawback of these models is their lack of generalizability across different domains. Some works [31, 38] employ meteorological principles and mathematical methods to simulate the chemical and physical processes of pollutants, facilitating prediction. While these models are underpinned by robust theoretical foundations, their specificity to certain domains poses a significant limitation in adaptability to diverse scenarios. 2) Gaussian process (GP) based models, which estimate unknown data points through weighted averages of known values [9]. GP applies the idea of random variables to the unknown as well as the values of the neighbor samples [27]. The unbiasedness and minimum variance constraints ultimately result in solving an optimization problem by adding constraints [30]. GP methods have shown success in scenarios with strong correlations among data sources [7, 21]. These methods require manual selection of distance metrics and kernel functions, parameterized by hyperparameters, to construct a covariance matrix. The limitation of GP lies in its intrinsic hypothesis in terms of the theoretical variogram and constant in mean assumption. The computational intensity and generation of the unsmooth surface also make it hardly generalizable [37]. The simplicity of the kernel equation often falls short in capturing complex nonlinear interactions. Numerical computation errors can lead to ill-conditioned matrices, hindering the accurate solution. Recent advancements, such as NARGP [28], combine machine learning with GP for enhanced flexibility and fit, representing the current state-of-the-art in GP methods for the multi-source prediction problems [2, 5, 6]. 3) Machine learning models. Compared to traditional and GP methods, machine learning models have gained popularity as a more flexible and computationally efficient approach for learning regression formulas for auxiliary attributes. With increasingly sophisticated architectures, machine learning models have become adept at capturing complex data correlations. For instance, Random Forests have been applied to multi-source spatial problems, utilizing geographical and ancillary features to build decision trees for prediction [11, 14, 41]. Nonetheless, these models often fall short in explicitly learning relationships between spatial points from different sources, limiting their effectiveness in multi-source

problems. The trend of combining multiple models to leverage their respective strengths and mitigate limitations is gaining traction [19, 20]. In spatial relationship studies, methods like Space2Vec [26] encode absolute positions and spatial relationships, emerging as state-of-the-art in spatial prediction [22, 25].

3 Problem Formulation

We formalize a point in a 2D geographic space as $s \in \mathbb{R}^2$ and consider N distinct data sets $\{D^{(i)}\}_{i=1}^{N}$, encompassing varied sources like ground stations and mini sensors. Each set $D^{(i)}$ is characterized by a unique set of geo-locations $S^{(i)} = \{s_j^{(i)}\}_{j=1}^{n_i}$, where n_i is the number of locations in $D^{(i)}$. These datasets collectively represent multi-source data within the same geographical region, providing diverse perspectives and measurements of the environmental conditions. Each geo-location $s_j^{(i)}$ within dataset $D^{(i)}$ is linked with a point sample $(x_j^{(i)}, y_j^{(i)})$, where $x_j^{(i)} \in \mathbb{R}^{p_i}$ represents the ancillary attributes, and p_i is the count of these attributes for $D^{(i)}$. The target observations $\{y_j^{(i)}\}_{j=1}^{n_i}$ in $D^{(i)}$ are seen as observed samples from the corresponding random variable $Y^{(i)}$, which is an approximation of the ground truth target variable $\tilde{Y}$. Thus, each dataset is represented as $D^{(i)} = \{(x_j^{(i)}, y_j^{(i)} | s_j^{(i)})\}_{j=1}^{n_i}$. We denote $y^{(i)} \sim Y^{(i)}$ as a general sampling from $Y^{(i)}$ that may not be included in dataset D_i. With the preliminary notions, we formalize the multi-source spatial point data prediction problem as: *Given multiple sets of data* $\{D^{(i)}\}_{i=1}^{N}$, *predict the target variable* $\tilde{y} \sim \tilde{Y}$ *across the 2D geospace.* In our model, for any given random location within the geospatial domain, the prediction generated is denoted as $\hat{y}$, correlating with a random variable $\hat{Y}$ such that $\hat{y} \sim \hat{Y}$. We omit the index for the prediction, reflecting its general applicability across the geospace. Fundamentally, the core objective of our model can be interpreted as maximizing the mutual information between random variables $\hat{Y}$ and $\tilde{Y}$, as it would enable our model to make highly accurate predictions by precisely mapping the predicted values to the actual, albeit unobserved, ground truth values.

This task faces several key challenges: 1) Aligning without supervision. The absence of ground truth makes it challenging to supervise and align the predictive mapping from various data sources to the prediction target. It is impossible to simply average across different sources due to their different measurement locations and quality, while it is also difficult to learn how to aggregate them because of the lack of ground truth. 2) Aligning across different qualities. The qualities of different sources are usually different yet difficult to quantify. Different data sources usually have different distributions of measurement errors, which are crucial to be considered while aggregating multi-source data. 3) Aligning across different locations. Different sources have different measurement locations so fusing such data requires not only aggregating across sources but also geolocations.

4 Methodology

To make accurate predictions by addressing the challenges outlined above, we propose the Deep Multi-source Spatial Prediction (DMSP) framework, as shown in Figure 2. This framework is designed to align information from diverse data sources. The alignment process is grounded in a self-supervised manner, which is derived from maximizing mutual information. The derivation and details of this approach are thoroughly discussed in Section 4.1. A key component in assessing the quality of each data source within this framework is the concept of a fidelity score. This score quantitatively represents the quality of each data source. We delve into the specifics of the fidelity score, including its definition and role in the training procedure, in Section 4.2. This section also offers a brief overview of the DMSP framework. In Section 4.3, we focus on the intricate details of the DMSP framework. Specifically, we explore the individual components that constitute DMSP, including independent graph convolution operators, a shared spatial relationship encoder, and a shared decoder.

4.1 Adaptive Self-supervised Fusion of Multiple Data Sources

As mentioned in the problem formulation section our objective is maximizing the mutual information between $\hat{Y}$ and $\tilde{Y}$. However, a crucial challenge arises from the unavailability of the ground truth $\tilde{Y}$. To address this, we propose a new idea. Specifically, we only leverage those ground truth values that are well observed by any sensors and by definition of weighted mutual information:

$$MI(\tilde{Y}, \hat{Y}) = \sum_{\hat{y} \in \hat{Y}} \sum_{\tilde{y} \in \tilde{Y}} w(\tilde{y}, \hat{y}) p(\tilde{y}, \hat{y}) \log \frac{p(\tilde{y}, \hat{y})}{p(\tilde{y}) p(\hat{y})}, \tag{1}$$

where $w(\tilde{y}, \hat{y})$ is the weight for generalized mutual information. We quantify $w(\tilde{y}, \hat{y})$ by the probability of being well estimated by each $y^{(i)}$, and in the extreme case, we only consider the case that the observation equals the ground truth:

$$\begin{aligned} MI(\tilde{Y}, \hat{Y}) &= \sum_{i=1}^{N} \sum_{\hat{y} \in \hat{Y}} \sum_{\tilde{y} \in \tilde{Y}} \mathbb{1}(y^{(i)} = \tilde{y}) p(\tilde{y}, \hat{y}) \log \frac{p(\tilde{y}, \hat{y})}{p(\tilde{y}) p(\hat{y})} \\ &= \sum_{i=1}^{N} \sum_{\hat{y} \in \hat{Y}} \sum_{y^{(i)} \in Y^{(i)}} \mathbb{1}(y^{(i)} = \tilde{y}) p(y^{(i)}, \hat{y}) \log \frac{p(y^{(i)}, \hat{y})}{p(y^{(i)}) p(\hat{y})}, \end{aligned} \tag{2}$$

where $\mathbb{1}$ is an indicator function. To generalize to the continuous case, we may use a soft probability to replace the indicator function and have:

$$\sum_{i=1}^{N} \sum_{\hat{y} \in \hat{Y}} \sum_{y^{(i)} \in Y^{(i)}} p(y^{(i)} = \tilde{y}) p(y^{(i)}, \hat{y}) \log \frac{p(y^{(i)}, \hat{y})}{p(y^{(i)}) p(\hat{y})}, \tag{3}$$

Here, the probability $p(y^{(i)} = \tilde{y})$ quantifies the overlap between random variables $Y^{(i)}$ and $\tilde{Y}$. Because each data source has its consistent measurement error, we can consider this probability as a constant C_i.

$$\sum_{i=1}^{N} \sum_{\hat{y} \in \hat{Y}} \sum_{y^{(i)} \in Y^{(i)}} C_i p(y^{(i)}, \hat{y}) \log \frac{p(y^{(i)}, \hat{y})}{p(y^{(i)}) p(\hat{y})} = \sum_{i=1}^{N} C_i MI(Y^{(i)}, \hat{Y}), \tag{4}$$

This equation represents our innovative approach to aligning data from multiple sources in an unsupervised setting. The observations from different data sources serve as approximations, enabling us to

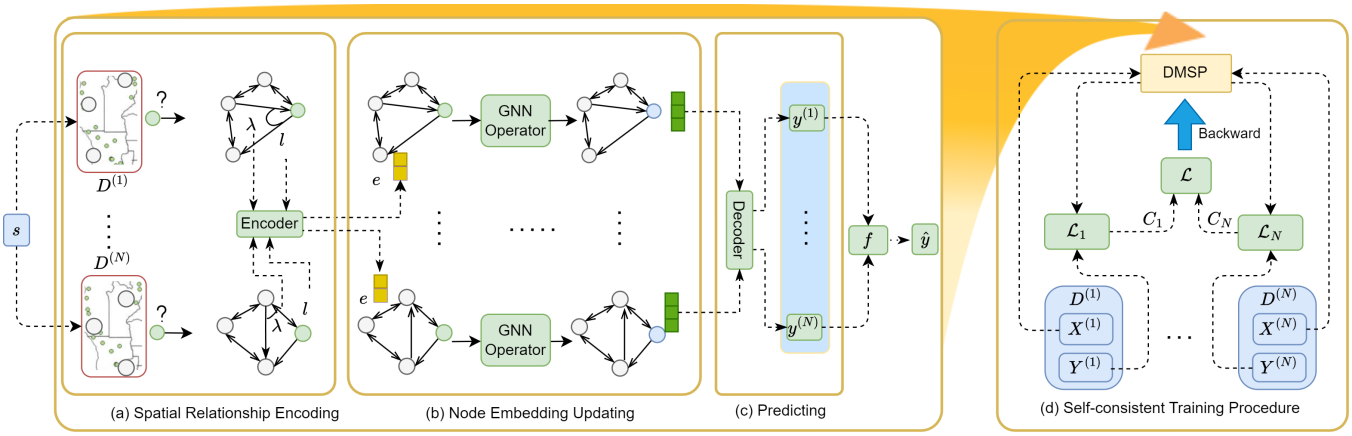

Figure 2: Illustration of the DMSP framework. For a target location s, the process involves: (a) building a KNN graph for each data source and using a shared spatial relationship encoder for spatial representations as edge features; (b) applying distinct graph convolution operators for updating node representations per data source; (c) employing a shared decoder for outputting predictions, which are fused by f for the final prediction. (d) shows the self-consistent training procedure of the framework.

harness their correlation with the ground truth target variable.

$$\begin{aligned} &\max \sum_{i=1}^{N} C_i MI(Y^{(i)}, \hat{Y}) = \sum_{i=1}^{N} C_i \left(H(Y^{(i)}) - H(Y^{(i)}|\hat{Y}) \right) \\ \Leftrightarrow &\max - \sum_{i=1}^{N} C_i H(Y^{(i)}|\hat{Y}) = \sum_{i=1}^{N} C_i \mathbb{E}_{p(y^{(i)},\hat{y})} \log p(y^{(i)}|\hat{y}), \end{aligned} \tag{5}$$

In Equation 5, the entropy term $H(Y^{(i)})$ is a constant for each dataset. Thus, maximizing mutual information essentially translates to maximizing the weighted sum of the expectation of the log conditional likelihood, which measures the uncertainty of the target random variable $Y^{(i)}$ given the model estimation. An ideal model should precisely determine the target variable. The actual conditional distribution p in Equation 5 is unknown. However, we may let the prediction model learn a distribution q that approximates the true conditional distribution p. According to the non-negativity of KL divergence, we can demonstrate that the conditional entropy $H(Y^{(i)}|\hat{Y})$ is minimized when the estimated conditional distribution aligns with the actual distribution. By approximating p with the learned distribution q, we establish a lower bound on the expectation of the log conditional likelihood as $\mathbb{E}_p \log q(y^{(i)}|\hat{y})$

Theorem 4.1. *For two random variables $Y^{(i)}, \hat{Y} \in \mathbb{R}$, any variational approximation of the conditional distribution $p_{Y^{(i)}|\hat{Y}}$ will increase the conditional entropy $H(Y^{(i)}|\hat{Y})$.*

This theorem supports our strategy of maximizing the lower bound as a means to achieve the objective in Equation 5. The optimization problem, as initially formulated in Equation 1, then becomes:

$$\max \sum_{i=1}^{N} C_i \mathbb{E}_{p(y^{(i)},\hat{y})} \log q(y^{(i)}|\hat{y}). \tag{6}$$

In this framework, the discrepancy between the estimation $\hat{Y}$ and the target variable $Y^{(i)}$ is characterized by a chosen distribution, such as Gaussian noise. Assuming the learned distribution q follows a normal distribution: $q \sim \mathcal{N}(\hat{Y}, \mathbb{E}_p(Y^{(i)} - \hat{Y})^2)$. Then Equation 6 is equivalent to minimizing a weighted square error loss. In a general case, we have:

$$\min \sum_{i=1}^{N} C_i \mathcal{L}_i(Y^{(i)}, \hat{Y}), \tag{7}$$

where $\mathcal{L}_i(Y^{(i)}, \hat{Y})$ is the tailored loss function for the two random variables. This formulation allows for flexibility in assuming any appropriate noise distribution as per the specific requirements of the problem. The detailed training process to achieve the objective outlined in Equation 7 is elaborated in Section 4.2.

4.2 Training Procedure and Framework

4.2.1 Self-consistent Training Procedure. As formulated in Equation 7, we transform the mutual information objective into a calculable function suitable for self-supervised training. Specifically, although direct access to ground truth values is not available, we may use target variable observations from all data sources as proxies for training the model. Notably, each loss term in Equation 7 is weighted by a constant C_i. While we have some prior knowledge about the relative quality of different data sources, the exact values of these constants are unknown. We define these as fidelity scores for each data source and treat them as learnable parameters. The self-consistent training procedure is designed to align model predictions with observations from all data sources by iteratively masking the target variable. To be specific, for a given sample $(x_j^{(i)}, y_j^{(i)}|s_j^{(i)})$ from dataset $D^{(i)}$, we mask the target variable $y_j^{(i)}$ and obtain a partially masked $D'^{(i)}$. Then the DMSP framework Φ_W estimates $y_j^{(i)}$ based on the collection of datasets $\{D^{(1)}, \ldots, D'^{(i)}, \ldots, D^{(N)}\}$, where W represents all parameters of DMSP. The loss $\mathcal{L}$ is calculated based on the prediction and $y_j^{(i)}$ according to the specific distribution assumption. To ensure meaningful fidelity scores and prevent them from becoming zero or negative, we impose two constraints: firstly, the fidelity score must be non-negative; secondly, the sum of all fidelity scores should total a constant, chosen to be 1

for simplicity. The modified objective in Equation 7 is thus:

$$\min_{W;\{C_i\}_{i=1}^N} \sum_{i=1}^{N} \sum_{j=1}^{n_i} C_i \mathcal{L}_i(y_j^{(i)}, \Phi_W) \text{ s.t. } 0 \le C_i, \sum_{i=1}^{N} C_i = 1. \quad (8)$$

Given the two constraints, we cannot optimize the parameters using standard gradient descent. However, we address this by redefining the fidelity score C_i as a function of an unconstrained variable $\overline{C}_i$, where $C_i = \frac{\exp \overline{C}_i}{\sum_{k=1}^{N} \exp \overline{C}_k}$.

Theorem 4.2. *Optimizing the unconstrained variables $\{\overline{C}_i\}_{i=1}^N$ is equivalent to optimizing the constrained variables $\{C_i\}_{i=1}^N$.*

This theorem allows us to transform our constrained optimization problem into an unconstrained one:

$$\min_{W;\{\overline{C}_{i=1}^N\}} \sum\nolimits_{i=1}^{N} \sum\nolimits_{j=1}^{n_i} \frac{\exp \overline{C}_i}{\sum_{k=1}^{N} \exp \overline{C}_k} \mathcal{L}(y_j^{(i)}, \Phi_W). \quad (9)$$

The details of the self-consistent training procedure are illustrated in Algorithm 1. Line 1 iterates through all datasets. Line 2 iterates through all samples of the current dataset. Line 3 gets the j-th sample. Line 4 masks the target value in the sample to obtain a temporal dataset. Line 5 predicts the masked target value using the proposed DMSP framework. Line 6 calculates the training loss. Line 7 to Line 10 updates the fidelity scores and the parameters of DMSP.

Algorithm 1: Single epoch training procedure

Input: $\{D^{(i)}\}_{i=1}^N$, initialized $\{C_i\}_{i=1}^N$, initialized Φ_W, Learning rate α.
Output: $\{C_i\}_{i=1}^N, \Phi_W$.

1 **for** $i \in \{1, \ldots, N\}$ **do**
2 **for** $j \in \{1, \ldots, n_i\}$ **do**
3 Get $(x_j^{(i)}, y_j^{(i)} | s_j^{(i)})$;
4 $D_i' \leftarrow D^{(i)}$ with $y_j^{(i)}$ masked;
5 $\hat{y} \leftarrow \Phi_W(D^{(1)}, \ldots, D'^{(i)}, \ldots, D^{(N)})$;
6 $\mathcal{L} \leftarrow C_i \mathcal{L}_i(\hat{y}, y_j^{(i)})$;
7 **for** $C \in \{C_1, \ldots, C_N\}$ **do**
8 $C \leftarrow C - \alpha \frac{\partial \mathcal{L}}{\partial C}$;
9 **end**
10 $W \leftarrow W - \alpha \frac{\partial \mathcal{L}}{\partial W}$;
11 **end**
12 **end**
13 **return** $\{C_i\}_{i=1}^N, \Phi_W$

4.2.2 Deep Multi-source Spatial Prediction. To make predictions by fusing information from all data sources, we propose the DMSP framework as shown in Figure 2. For a given target location s, we first construct a K-Nearest Neighbor (KNN) graph for each data source based on sample geo-locations (detailed in Section 4.3.1). To consider the influences of geolocation environments, we propose a spatial relationship encoder ζ_{SD} (Figure 2 (a)) to extract the spatial relationships from the coordinate values. The encoder is shared across all data sources due to their shared geographic space (Figure 2 (a)). Details of the encoder are elaborated in Section 4.3.2.

After extracting spatial representations, distinct graph convolution operators are applied to each data source to fit their specific attribute and target variable relationships, as shown in Figure 2 (b). These operators facilitate specialized training for unified node embeddings, essential for target variable prediction (detailed in Section 4.3.3).

In Figure 2 (c), a shared decoder ζ_{SD}, which can be a multilayer perceptron network, transforms the final representation of the target node into the target variable prediction for each data source. Finally, the predictions from different sources are fused through a prediction function f. In practice, we choose to implement f as a weighted sum of all source predictions based on their fidelity scores: $\hat{y} = \sum_{i=1}^{N} C_i y^{(i)}$. In Figure 2 (d), fidelity scores are utilized to calculate weighted losses during training with target variables from different sources. By utilizing a gradient descent algorithm, our framework optimizes towards the objective in Equation 7.

4.3 Geo-location-aware Graph Neural Networks

In this section, we introduce the construction of the input graph for prediction and the integration of spatial information into the graph neural network within the DMSP framework.

4.3.1 Spatial K-Nearest Neighbor Graph. Spatial data inherently exhibits spatial correlation, where nearby points often share similar attributes. To harness this characteristic, we construct a K-nearest neighbor (KNN) graph based on sample locations. For a sample $(x_j^{(i)}, y_j^{(i)} | s_j^{(i)})$ in data source $D^{(i)}$, with $s_j^{(i)}$ representing the locations of simultaneous samples, we build a KNN graph $G_j^{(i)}$ using $s_j^{(i)}$ and $s_j^{(i)}$. It's important to note that samples within the same data source may have different timestamps, leading to a varying number of samples ($n_{i,j}$) at a given time. Thus $n_{i,j}$ is less than the total sample number n_i in data source $D^{(i)}$.

The graph $G_j^{(i)} = (V_j^{(i)}, E_j^{(i)}, \overline{x}_j^{(i)})$ consists of a node set $V_j^{(i)}$, which is the union of the target location $s_j^{(i)}$ and $s_j^{(i)}$, and edge feature set $E_j^{(i)} \subseteq V_j^{(i)} \times V_j^{(i)}$ denoting the relationships between locations. The edge feature will be detailed in Section 4.3.2. Directed edges connect the K-nearest neighbors to the central node. Therefore, there are exactly $K \| V_j^{(i)} \|$ edges in the graph. The node attribute matrix $\overline{x}_j^{(i)} \in \mathbb{R}^{n_{i,j} \times (p_i+1)}$ corresponds to features from the associated samples.

4.3.2 Spatial Relationship Encoder. Capturing spatial relationships between nodes in a graph requires more than just using coordinate values or distances, as these methods often fall short in integrating comprehensive spatial information. Coordinates can impose unnecessary constraints and fail to recognize simple translations or rotations within the dataset. Similarly, distances alone cannot represent the orientation of neighboring nodes or account for the combined effects of multiple neighboring nodes. Recognizing these limitations, we adopt a more holistic approach.

Instead of relying on a simplistic distance-based adjacency matrix, our approach jointly considers distances and angles to represent node relationships. We utilize a spatial relationship encoder ζ_{SE} to dynamically compute the relationships between nodes based on both distance and angular measurements. For each edge $(v_{j;k}^{(i)}, v_{j;o}^{(i)})$ in the graph, where $v_{j;o}^{(i)}$ is the target node and $v_{j;k}^{(i)}$ is a source node, we define an edge representation $e_{j;k,o}^{(i)} \in E_j^{(i)}$ as follows:

$$e_{j;k,o}^{(i)} = \zeta_{SE}\left(l_{j;o,k}^{(i)}, \lambda_{j;o,k}^{(i)}\right), \tag{10}$$

where $l_{j;o,k}^{(i)}$ is the distance between the two nodes, and $\lambda_{j;o,k}^{(i)} \in [-\pi, \pi)$ is the angle at node $v_{j;k}^{(i)}$ formed by rotating clockwise from the ray $\overrightarrow{v_{j;k}^{(i)} v_{j;o}^{(i)}}$ until it aligns with the direction of the next neighboring node of $v_{j;k}^{(i)}$. This representation is generalized from [39]. Compared to the original representation, we have a relaxed requirement for the number of samples as we do multi-source prediction.

4.3.3 Graph Convolution Operators. To derive meaningful node embeddings that capture attribute-target relationships by incorporating the computed spatial representations, we employ multiple layers of graph convolutional operators. These operators process both node and edge features. The updated node attributes are expressed as $\overline{x}_j^{(i)(l+1)} = g^{(i)(l)}(\overline{x}_j^{(i)(l)}, E_j^{(i)})$, where $\overline{x}_j^{(i)(l+1)}$ represents the updated node attributes after l layers, and $g^{(i)(l)}$ is the graph convolution operator at layer l for graphs constructed from data source $D^{(i)}$. It's important to note that these graph convolutional operators are distinct for each data source, allowing for customized processing of different data characteristics. After applying L layers of graph convolution, the embedding of the target node is $x_{j;k}^{(i)(L)}$, which is passed to the downstream decoder to make the final prediction. The dynamic nature of the graph $G_j^{(i)}$, which changes over time and adapts to varying sensor data, enables the model to learn inductively, without reliance on a fixed graph structure.

5 Experiment

In this section, we first introduce the experimental settings, then we evaluate the effectiveness and robustness of the proposed method and show the performance result against comparison methods. After that, we perform ablation studies to verify the contribution of different parts of the proposed framework. Finally, we show some analysis of scalability and the sensitivity of hyperparameters to help further explore the characteristics of the proposed framework. The implementation details and efficiency analysis can be found in the appendix. All the experiments are conducted on a 64-bit machine with an NVIDIA A5000 GPU.

5.1 Experiment Setting

5.1.1 Dataset. 1) **SouthCalAir**[33]: Fine particulate matter (PM 2.5) data from 26 Environmental Protection Agency's (EPA) Air Quality System (AQS) sensors and 515 PurpleAir sensors in south California [34] from 2019/01/01 to 2019/12/31. EPA AQS data are considered high-quality while PurpleAir data are noisy. 2) **NorthCalAir**[33]: Similar to the SouthCalAir dataset, PM 2.5 data from 63 Environmental Protection Agency's (EPA) Air Quality System (AQS) sensors and 1110 PurpleAir sensors in north California from 2019/01/01 to 2019/12/31. 3) **Spatially correlated regression (SCR)**: A synthetic dataset generated by simulating spatially correlated data that involves creating features distributed across space with spatial correlations [12] and simulating non-linear relationships among features and the target variable [16]. Gaussian normal noise is added to the target variable to simulate the low-quality data source. Features are irreversibly transformed to avoid degradation to a simple regression task. 4) **Flu**: The Flu dataset integrates data from two distinct sources: the Centers for Disease Control and Prevention (CDC) and the Google Flu Trends program [8] from 2010 to 2015. The CDC provides data from medical providers, which is considered more reliable but is limited in geographical coverage (48 states). Conversely, Google Flu Trends estimates activity levels based on public keyword searches, offering broader coverage but less reliability. The results of this dataset can be found in the appendix.

5.1.2 Comparison Method. 1) **SRA-MLP** [20]: A model that combines the stepwise regression analysis and multilayer perceptron neural network. 2) **RR-XGBoost** [19]: A model that combines ridge regression and the Extreme Gradient Boosting Algorithm. 3) **NARGP** [28]: A probabilistic framework based on Gaussian process regression and non-linear autoregressive schemes. 4) **GeoPrior** [23]: An efficient spatial prior is proposed in the method to estimate the probability conditioned on a geographical location. 5) **Space2Vec** [26]: A representation learning model to encode the absolute positions and spatial relationships of places.

5.2 Effectiveness Analysis

5.2.1 Evaluation Metrics. To quantify the effectiveness of the models, we use the following five metrics: 1) Mean absolute error (MAE), 2) Root-mean-square error (RMSE), 3) Explained variance score (EVS), 4) Coefficient of determination (CoD), 5) Pearson correlation coefficient (Pearson).

5.2.2 Performance. Table 1 summarized the performance comparison among the proposed methods and competing models on the real-world and synthetic datasets. The results are obtained from 3 individual runs for every setting. We report the standard deviation over three runs following the ± mark. The best result for each dataset is highlighted with boldface font and the second best is underlined. In general, our proposed framework outperformed comparison methods in terms of all the evaluation metrics. Note that the GeoPrior and Space2Vec can be included in our framework to substitute the Geo-location-aware graph neural network part, so we include them as ablation study results in this table, together with two other ablation models DMSP-H and DMSP-F, which will be introduced later. It is apparent from the results that the proposed method excels over other methods and achieves the best performance on all the datasets. For instance, DMSP reduced the MAE error by 27.5% compared to the average prediction of other methods for the air pollutant prediction task on the NorthCalAir dataset, and by 19.2% compared to the second-best model. What's more, DMSP

Table 1: The performance of the proposed model (including ablation variants) and the comparison methods.

Dataset	Method	MAE	RMSE	EVS	CoD	Pearson
SouthCalAir	SRA-MLP	3.211±0.059	5.305±0.091	0.416±0.017	0.411±0.015	0.686±0.012
	RR-XGBoost	5.811±0.047	8.450±0.107	-0.258±0.050	-0.2644±0.056	0.351±0.010
	NARGP	4.476±0.853	7.000±1.484	0.084±0.334	0.076±0.331	0.487±0.138
	DMSP	**3.112±0.059**	**4.878±0.234**	**0.542±0.026**	**0.504±0.036**	**0.737±0.021**
	GeoPrior	3.236±0.305	5.926±0.254	0.411±0.107	0.411±0.109	0.670±0.041
	Space2Vec	3.135±0.303	5.996±0.283	0.401±0.122	0.395±0.107	0.672±0.056
	DMSP-H	4.091±0.486	6.106±0.344	0.277±0.119	0.263±0.139	0.555±0.059
	DMSP-F	14.835±2.850	22.445±3.768	-6.972±2.271	-9.361±2.379	0.055±0.074
NorthCalAir	SRA-MLP	3.000±0.059	5.374±0.378	0.404±0.026	0.390±0.024	0.636±0.021
	RR-XGBoost	3.705±0.030	6.177±0.334	0.121±0.066	0.119±0.068	0.557±0.014
	NARGP	3.317±0.035	6.26±0.580	0.186±0.158	0.185±0.158	0.579±0.052
	DMSP	**2.423±0.083**	**4.474±0.489**	**0.590±0.052**	**0.586±0.046**	**0.768±0.034**
	GeoPrior	2.845±0.055	4.920±0.154	0.481±0.027	0.480±0.029	0.690±0.021
	Space2Vec	2.641±0.103	4.509±0.283	0.585±0.022	0.584±0.027	0.762±0.056
	DMSP-H	3.768±0.823	6.048±1.394	0.236±0.274	0.221±0.286	0.506±0.205
	DMSP-F	17.110±3.289	42.426±20.589	-32.214±33.699	-37.378±35.118	0.003±0.013
SCR	SRA-MLP	0.782±0.031	0.969±0.029	0.007±0.008	-0.016±0.014	0.092±0.080
	RR-XGBoost	0.939±0.014	1.208±0.032	-0.573±0.177	-0.646±0.236	0.031±0.040
	NARGP	0.616±0.045	0.773±0.059	0.457±0.024	0.446±0.025	0.698±0.012
	DMSP	**0.478±0.032**	**0.574±0.025**	**0.606±0.046**	**0.605±0.045**	**0.780±0.027**
	GeoPrior	0.505±0.035	0.600±0.054	0.553±0.037	0.553±0.039	0.731±0.021
	Space2Vec	0.498±0.015	0.615±0.063	0.584±0.031	0.580±0.029	0.751±0.046
	DMSP-H	0.516±0.026	0.631±0.030	0.507±0.135	0.489±0.158	0.726±0.071
	DMSP-F	0.484±0.018	0.609±0.014	0.596±0.048	0.588±0.047	0.776±0.026

achieves a Pearson score of 0.768 while the second-best score is only 0.636 on the NorthCalAir dataset. For the comparison methods, Even though they can achieve similar scores sometimes, their prediction performance may not be consistent on all the datasets. For example, SRA-MLP delivers the second-best scores on the SouthCalAir and NorthCalAir datasets, but it performs badly on the SCR dataset. This is due to the fact that the target value in the SCR dataset can not be determined by the sample features. SRA-MLP lacks the ability to predict based on the information of neighbors by learning spatial relationships. The GP-based SOTA NARGP has a consistent performance on all the datasets, but the overall performance is not comparable to DMSP. This implies that NARGP has the ability to model spatial correlations and predict according to neighbor samples. However, since it relies on fixed kernels, it fails to accurately model the spatial relationships in the dataset.

5.3 Ablation Study

We conduct three ablation studies here. We first investigate the benefit gained from the proposed objective function in Equation 8. Specifically, we 1) examine whether a single data source has enough information; and 2) examine the impact of treating each data source equally. Then we investigate the effectiveness of the proposed graph neural network by 3) comparing it with two SOTA spatial embedding methods.

5.3.1 Information from a Single Source. For the first question, we propose to use the target variable in a single source as supervision in the loss calculation. The best result on a single data source is reported and we name this variation of our model DMSP-H. The results on all the datasets are shown in Table 1. Overall, the performance will degrade if the model only learns from a single data source. Particularly, the performance drop is most significant on the NorthCalAir dataset, and the DMSP-H becomes the worst among the comparison methods. This is due to the fact that it is the largest dataset where each source provides a significant contribution to the final prediction. This study shows that although a single source cannot provide enough information to achieve the best performance.

5.3.2 Mixed Multiple Sources. We further conduct another ablation study where we remove the use of fidelity score and treat all the data sources equally. We name this model DMSP-F and the performances are shown in Table 1. We notice a huge performance drop on the two air pollutants datasets. This implies that the low-quality sources in the air pollutants datasets are extremely noisy and provide misleading guidance for the whole model. Notably, the performance on the SCR dataset almost does not change. This is due to the fact that we only add a zero-mean Gaussian random noise to the low-quality data source in the SCR dataset. So the general trend of the low-quality data source is correct and the neural network model is resistant to such an ideal noise. This also suggests that the noise in the real-world dataset is hard to model. This study shows the huge noise from real-world datasets and indicates that there is not always a benefit from fusing multiple data sources.

5.3.3 Different Spatial Embeddings in GNN. We substitute the proposed geo-location-aware graph neural networks by GeoPrior and Space2Vec. As we can observe, GeoPrior does not perform well. This is because the original coordinate values do not hold any meaningful information for the prediction problem. The Space2Vec also

fails to provide meaningful information for the spatial point prediction problem. In the point data prediction problem, the crucial geo factor is the relative position and angles between the target node and its neighbors. The two comparison methods aim to create a unique embedding for any location in the study area from a global perspective, similar to the positional encoder method used in language models like transformers.

5.4 Case Study

Here we further provide two case studies on the SouthCal and SCR datasets showing the predictions from each data source and the fused prediction compared with the observations from the high-fidelity data source as the approximation of the ground truth. The unconstrained fidelity scores are $\{14.837, -17.930\}$ for SouthCal and $\{2.836, -1.836\}$ for SCR. The result matches the prior knowledge that the low-quality data source is assigned a low score. As shown in Figure 3, in general, the prediction from the high-quality source is closer to the observed value, while the low-quality source prediction is not accurate. And the proposed framework can automatically learn the fidelity scores which assign the weight that can effectively safeguard against wrong estimations from the low-quality data source.

5.5 Scalability Analysis

To analyze the scalability of the proposed framework, we record the average runtime for all the methods. The runtime is shown with respect to the number of samples. We take the number of samples in the original SCR dataset as the counting unit and generate 10, 25, 50, and 100 times the number of samples. The results are presented in Figure 4. As we can see, all the machine learning methods demonstrate similar linear increase patterns with the growth of the number of samples. Our model has a slightly higher runtime than other machine learning methods, but they are still in the same order of magnitude. The test case is similar to the training one, where our model still shows linear growth with respect to the number of samples.

5.6 Sensitivity Analysis

Here we provide a sensitivity analysis of two hyperparameters in the proposed framework: the number of neighbors K chosen to build the KNN graph and the number of GNN layers. Figure 5 shows the performance for various values of the hyperparameters on the SCR dataset. The sensitivity of the number of neighbors is shown in Figure 5a. In general, the model is not very sensitive to the number of neighbors within the range studied. And it outperformed the baseline by a significant margin in terms of all the metrics. The optimal value for the number of neighbors is around 4. While the specific best number of neighbors can vary depending on the dataset as well as the task, in general, the proposed framework can perform well when the number is relatively small. The sensitivity of the number of GNN layers is shown in Figure 5b. The optimal value for the number of GNN layers is around 2.

6 Conclusion

We propose a new framework for multi-source spatial point data prediction. Central to this framework is a self-supervised training objective that not only aligns data from various sources but also leverages their collective strength without ground truth data. The fidelity score quantitatively evaluates the quality of each data source. Moreover, the geo-location-aware graph neural network adeptly handles the intricate spatial relationships across different locations. The efficacy of our methods has been rigorously tested and validated on both synthetic and real-world multi-source datasets. The results demonstrate that our framework not only outperforms existing state-of-the-art methods but also provides meaningful insights into the data.

Acknowledgments

This work was supported by the NSF Grant No. 2432418, No. 2414115, No. 2007716, No. 2007976, No. 1942594, No. 1907805, No. 2318831, Cisco Faculty Research Award, Amazon Research Award.

References

[1] Indoor Air Quality Act. 2011. Environmental Protection Administration: Taiwan.
[2] Kai Chen, Qinglei Kong, Yijue Dai, Yue Xu, Feng Yin, Lexi Xu, and Shuguang Cui. 2021. Recent advances in data-driven wireless communication using gaussian processes: A comprehensive survey. *arXiv preprint arXiv:2103.10134* (2021).
[3] Ling-Jyh Chen, Yao-Hua Ho, Hu-Cheng Lee, Hsuan-Cho Wu, Hao-Min Liu, Hsin-Hung Hsieh, Yu-Te Huang, and Shih-Chun Candice Lung. 2017. An open framework for participatory PM2. 5 monitoring in smart cities. *Ieee Access* 5 (2017), 14441–14454.
[4] Pi-Cheng Chen and Yu-Ting Lin. 2022. Exposure assessment of PM2. 5 using smart spatial interpolation on regulatory air quality stations with clustering of densely-deployed microsensors. *Environmental Pollution* 292 (2022), 118401.
[5] Sibo Cheng, César Quilodrán-Casas, Said Ouala, Alban Farchi, Che Liu, Pierre Tandeo, Ronan Fablet, Didier Lucor, Bertrand Iooss, Julien Brajard, et al. 2023. Machine learning with data assimilation and uncertainty quantification for dynamical systems: a review. *IEEE/CAA Journal of Automatica Sinica* 10, 6 (2023), 1361–1387.
[6] Dominik Cvetek, Mario Muštra, Niko Jelušić, and Leo Tišljarić. 2021. A survey of methods and technologies for congestion estimation based on multisource data fusion. *Applied Sciences* 11, 5 (2021), 2306.
[7] Volker L Deringer, Albert P Bartók, Noam Bernstein, David M Wilkins, Michele Ceriotti, and Gábor Csányi. 2021. Gaussian process regression for materials and molecules. *Chemical Reviews* 121, 16 (2021), 10073–10141.
[8] Andrea Freyer Dugas, Mehdi Jalalpour, Yulia Gel, Scott Levin, Fred Torcaso, Takeru Igusa, and Richard E Rothman. 2013. Influenza forecasting with Google flu trends. *PloS one* 8, 2 (2013), e56176.
[9] Paul Daniel Dumitru, Marin Plopeanu, and Dragos Badea. 2013. Comparative study regarding the methods of interpolation. *Recent advances in geodesy and Geomatics engineering* 1 (2013), 45–52.
[10] Sumanth Reddy Enigella and Hamid Shahnasser. 2018. Real time air quality monitoring. In *2018 10th International Conference on Knowledge and Smart Technology (KST)*. IEEE, 182–185.
[11] Guannan Geng, Xia Meng, Kebin He, and Yang Liu. 2020. Random forest models for PM2. 5 speciation concentrations using MISR fractional AODs. *Environmental Research Letters* 15, 3 (2020), 034056.
[12] Arthur Getis. 2009. Spatial autocorrelation. In *Handbook of applied spatial analysis: Software tools, methods and applications*. Springer, 255–278.
[13] Jay K Hackett and Mubarak Shah. 1990. Multi-sensor fusion: a perspective. In *Proceedings., IEEE International Conference on Robotics and Automation*. IEEE, 1324–1330.
[14] Xuefei Hu, Jessica H Belle, Xia Meng, Avani Wildani, Lance A Waller, Matthew J Strickland, and Yang Liu. 2017. Estimating PM2. 5 concentrations in the conterminous United States using the random forest approach. *Environmental science & technology* 51, 12 (2017), 6936–6944.
[15] Zhe Jiang. 2018. A survey on spatial prediction methods. *IEEE Transactions on Knowledge and Data Engineering* 31, 9 (2018), 1645–1664.
[16] Yee Leung, Chang-Lin Mei, and Wen-Xiu Zhang. 2000. Statistical tests for spatial nonstationarity based on the geographically weighted regression model. *Environment and Planning A* 32, 1 (2000), 9–32.
[17] Jin Li and Andrew D Heap. 2014. Spatial interpolation methods applied in the environmental sciences: A review. *Environmental Modelling & Software* 53 (2014), 173–189.
[18] Bing Liu, Yueqiang Jin, Dezhi Xu, Yishu Wang, and Chaoyang Li. 2021. A data calibration method for micro air quality detectors based on a LASSO regression and NARX neural network combined model. *Scientific Reports* 11, 1 (2021), 21173.

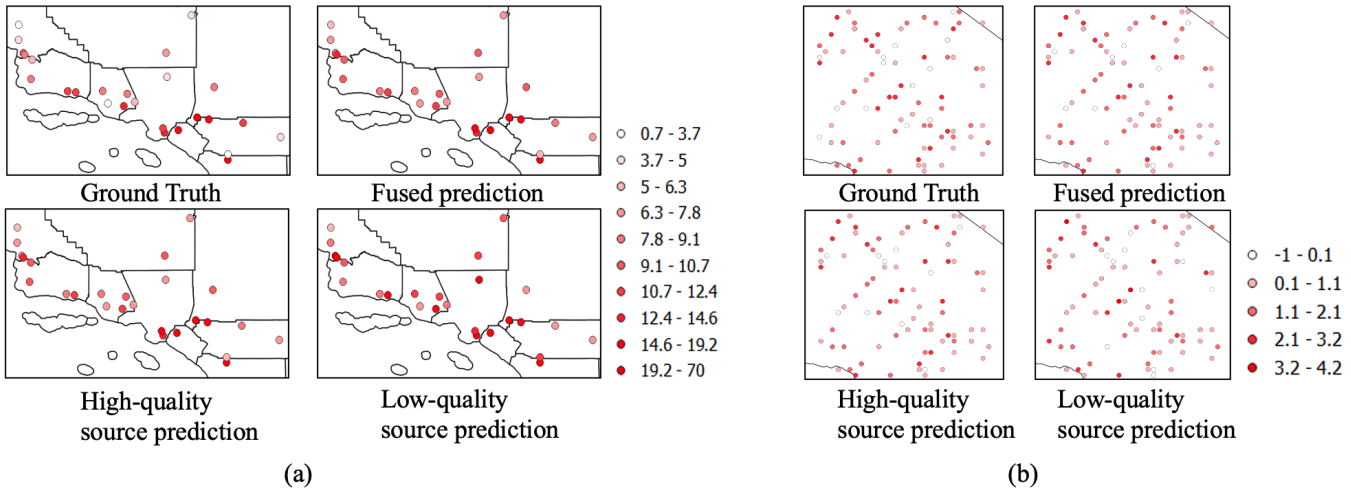

Figure 3: Visualization for the prediction and approximate ground truth for (a) SouthCal and (b) SCR

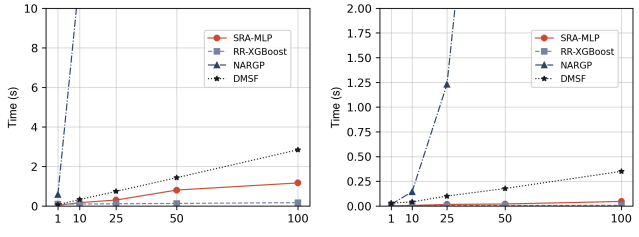

Figure 4: The scalability study about the number of samples.

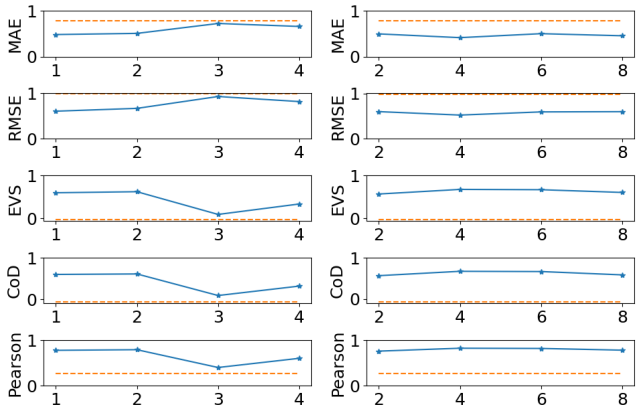

Figure 5: The sensitivity study of two hyperparameters in DMSP on the SCR dataset. The dashed lines represent the average baseline performance.

[19] Bing Liu, Xianghua Tan, Yueqiang Jin, Wangwang Yu, and Chaoyang Li. 2021. Application of RR-XGBoost combined model in data calibration of micro air quality detector. *Scientific Reports* 11, 1 (2021), 1–14.

[20] Bing Liu, Qingbo Zhao, Yueqiang Jin, Jiayu Shen, and Chaoyang Li. 2021. Application of combined model of stepwise regression analysis and artificial neural network in data calibration of miniature air quality detector. *Scientific Reports* 11, 1 (2021), 3247.

[21] Haitao Liu, Yew-Soon Ong, Xiaobo Shen, and Jianfei Cai. 2020. When Gaussian process meets big data: A review of scalable GPs. *IEEE transactions on neural networks and learning systems* 31, 11 (2020), 4405–4423.

[22] Pengyuan Liu and Filip Biljecki. 2022. A review of spatially-explicit GeoAI applications in Urban Geography. *International Journal of Applied Earth Observation and Geoinformation* 112 (2022), 102936.

[23] Oisin Mac Aodha, Elijah Cole, and Pietro Perona. 2019. Presence-only geographical priors for fine-grained image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9596–9606.

[24] Sachit Mahajan, Cyuan-Heng Luo, Dong-Yi Wu, and Ling-Jyh Chen. 2021. From Do-It-Yourself (DIY) to Do-It-Together (DIT): Reflections on designing a citizen-driven air quality monitoring framework in Taiwan. *Sustainable Cities and Society* 66 (2021), 102628.

[25] Gengchen Mai, Weiming Huang, Jin Sun, Suhang Song, Deepak Mishra, Ninghao Liu, Song Gao, Tianming Liu, Gao Cong, Yingjie Hu, et al. 2023. On the opportunities and challenges of foundation models for geospatial artificial intelligence. *arXiv preprint arXiv:2304.06798* (2023).

[26] Gengchen Mai, Krzysztof Janowicz, Bo Yan, Rui Zhu, Ling Cai, and Ni Lao. 2020. Multi-scale representation learning for spatial feature distributions using grid cells. *arXiv preprint arXiv:2003.00824* (2020).

[27] Margaret A Oliver and Richard Webster. 1990. Kriging: a method of interpolation for geographical information systems. *International Journal of Geographical Information System* 4, 3 (1990), 313–332.

[28] Paris Perdikaris, Maziar Raissi, Andreas Damianou, Neil D Lawrence, and George Em Karniadakis. 2017. Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 473, 2198 (2017), 20160751.

[29] Michael Petroni, Dustin Hill, Lylla Younes, Liesl Barkman, Sarah Howard, I Brielle Howell, Jaime Mirowsky, and Mary B Collins. 2020. Hazardous air pollutant exposure as a contributing factor to COVID-19 mortality in the United States. *Environmental Research Letters* 15, 9 (2020), 0940a9.

[30] Je-Seon Ryu, Min-Soo Kim, Kyung-Joon Cha, Tae Hee Lee, and Dong-Hoon Choi. 2002. Kriging interpolation methods in geostatistics and DACE model. *KSME International Journal* 16, 5 (2002), 619–632.

[31] Jagriti Saini, Maitreyee Dutta, and Gonçalo Marques. 2020. A comprehensive review on indoor air quality monitoring systems for enhanced public health. *Sustainable environment research* 30, 1 (2020), 1–12.

[32] Christian Schmidt, Rohini Kumar, Soohyun Yang, and Olaf Büttner. 2020. Microplastic particle emission from wastewater treatment plant effluents into river networks in Germany: Loads, spatial patterns of concentrations and potential toxicity. *Science of the Total Environment* 737 (2020), 139544.

[33] Dharmendra Singh, Meenakshi Dahiya, Rahul Kumar, and Chintan Nanda. 2021. Sensors and systems for air quality assessment monitoring and management: A review. *Journal of environmental management* 289 (2021), 112510.

[34] Emily G Snyder, Timothy H Watkins, Paul A Solomon, Eben D Thoma, Ronald W Williams, Gayle SW Hagler, David Shelow, David A Hindin, Vasu J Kilaru, and

Peter W Preuss. 2013. The changing paradigm of air pollution monitoring. *Environmental science & technology* 47, 20 (2013), 11369–11377.
[35] Alexander Y Sun, Peishi Jiang, Zong-Liang Yang, Yangxinyu Xie, and Xingyuan Chen. 2022. A graph neural network (GNN) approach to basin-scale river network learning: the role of physics-based connectivity and data fusion. *Hydrology and Earth System Sciences* 26, 19 (2022), 5163–5184.
[36] Steven T Turnock, Robert J Allen, Martin Andrews, Susanne E Bauer, Makoto Deushi, Louisa Emmons, Peter Good, Larry Horowitz, Jasmin G John, Martine Michou, et al. 2020. Historical and future changes in air pollutants from CMIP6 models. *Atmospheric Chemistry and Physics* 20, 23 (2020), 14547–14579.
[37] Maleika Wojciech. 2018. Kriging method optimization for the process of DTM creation based on huge data sets obtained from MBESs. *Geosciences* 8, 12 (2018), 433.
[38] Xingzhe Xie, Ivana Semanjski, Sidharta Gautama, Evaggelia Tsiligianni, Nikos Deligiannis, Raj Thilak Rajan, Frank Pasveer, and Wilfried Philips. 2017. A review of urban air pollution monitoring and exposure assessment methods. *ISPRS International Journal of Geo-Information* 6, 12 (2017), 389.
[39] Dazhou Yu, Guangji Bai, Yun Li, and Liang Zhao. 2022. Deep Spatial Domain Generalization. In *2022 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1293–1298.
[40] Dazhou Yu, Binbin Chen, Yun Li, Suman Dhakal, Yifei Zhang, Zhenke Liu, Minxing Zhang, Jie Zhang, and Liang Zhao. 2024. STES: A Spatiotemporal Explanation Supervision Framework. In *Proceedings of the 2024 SIAM International Conference on Data Mining (SDM)*. SIAM, 73–81.
[41] Mehdi Zamani Joharestani, Chunxiang Cao, Xiliang Ni, Barjeece Bashir, and Somayeh Talebiesfandarani. 2019. PM2. 5 prediction based on random forest, XGBoost, and deep learning using multisource remote sensing data. *Atmosphere* 10, 7 (2019), 373.
[42] Minxing Zhang, Dazhou Yu, Yun Li, and Liang Zhao. 2022. Deep geometric neural network for spatial interpolation. In *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*. 1–4.
[43] Minxing Zhang, Dazhou Yu, Yun Li, and Liang Zhao. 2023. Deep Spatial Prediction via Heterogeneous Multi-source Self-supervision. *ACM Transactions on Spatial Algorithms and Systems* 9, 3 (2023), 1–26.

A Supplementary Material

A.1 Proof of Theorem 4.1

Proof. Let $H'(Y^{(i)}|\hat{Y})$ be the variational approximated conditional entropy,

$$
\begin{aligned}
H'(Y^{(i)}|\hat{Y}) &= -\mathbb{E}_p \log q(y^{(i)}|\hat{y}) \\
H(Y^{(i)}|\hat{Y}) &= -\mathbb{E}_p \log p(y^{(i)}|\hat{y}) \\
&= -\mathbb{E}_p \left[\log p(y^{(i)}|\hat{y}) + \log q(y^{(i)}|\hat{y}) - \log q(y^{(i)}|\hat{y})\right] \\
&= -\mathbb{E}_p \left[\log q(y^{(i)}|\hat{y}) + \log \frac{p(y^{(i)}|\hat{y})}{q(y^{(i)}|\hat{y})}\right] \\
&= -\mathbb{E}_p \log q(y^{(i)}|\hat{y}) - D_{\mathbb{KL}}(p \parallel q) \\
&\leq -\mathbb{E}_p \log q(y^{(i)}|\hat{y}) \\
&= H'(Y^{(i)}|\hat{Y})
\end{aligned}
$$

□

A.2 Proof of Theorem 4.2

The proof of this theorem is a consequence of the following two lemmas. **Lemma 1.** *$\{C_i\}_{i=1}^N$ obtained from $\{\overline{C}_i\}_{i=1}^N$ satisfies the original two constraints on C_i.*

Proof.

$$
0 \leq C_i = \frac{\exp \overline{C}_i}{\sum_{k=1}^N \exp \overline{C}_k} \leq 1,
$$

$$
\sum_{i=1}^N C_i = \frac{\sum_{i=1}^N \exp \overline{C}_i}{\sum_{k=1}^N \exp \overline{C}_k} = 1,
$$

□

Lemma 2. *For any optimal set $\{C_i^*\}_{i=1}^N$ from the original optimization problem, there exists $\{C_i\}_{i=1}^N$ equals $\{C_i^*\}_{i=1}^N$.*

Proof. Base case: For $C_1^* \in [0,1]$, we can have $C_1 = C_1^*$. Inductive hypothesis: Assume we have $\{C_i = C_i^*\}_{i=1}^n, n \geq 1$. Inductive step: Let $A = \sum_{i=1}^n C_i$. Then $C_{i+1}^* \in [0, 1-A]$, and $C_{i+1} \in [0, 1-A]$ can take any value in the range, so we can have $C_{i+1} = C_{i+1}^*$. By induction, we can have $\forall i \in \{1, \cdots, N\}, C_i = C_i^*$. □

Proof. As stated in Lemma 1 and 2, if there exists an optimal solution $\{C_i^*\}_{i=1}^N$ of the original optimization problem, we can get the same set that satisfies the original constraints via optimizing $\{\overline{C}_i\}_{i=1}^N$. And it's obvious that the set $\{C_i\}_{i=1}^N$ obtained from the optimal $\{\overline{C}_i^*\}_{i=1}^N$ of the new problem must also be optimal in the original problem. □

A.3 Implementation details

We assume a Gaussian normal noise for the data and choose the RMSE loss for $\mathcal{L}$. For the spatial feature encoder ζ_{SE} and the decoder ζ_{SD}, we leverage 3-layer MLPs. For each dataset, we randomly split it into 60%, 20%, and 20% for training, validation, and testing respectively. All the Deep learning models are trained for a maximum of 500 epochs using an early stop scheme. Adam optimizer with a learning rate of 0.001 is used. For the SRA-MLP method, we leveraged the LinearRegression model in sklearn package and implemented a multi-layer perceptron network using the PyTorch framework. The hyperparameters we tuned and their respective range were: dimension of hidden features in [32,64,128], the number of layers in [2,4,8,16]. The best hyperparameter we found were 128 and 8. For the RR-XGBoost method, we used the Ridge Regression model in the scikit-learn package and the xgboost package to implement the model. We used default values for most of the parameters and focused on two sensitive hyperparameters: number of estimators in [100,1000,10000] and max depth in [2,4,8]. We found the best hyperparameter were 1000 and 4. For the NARGP method, since the code is available we followed the original settings but increased the maximum iteration number to 2000 to ensure convergence. For our DMSP model, we used a 3-layer MLP for the decoder and a single-layer forward network for the spatial relationship encoder. We set the dimension of hidden embeddings in all the modules to be the same (16). The hyperparameters we tuned are the number of neighbors K in [1,3] and the number of GNN layers in [2,4,8]. We set K to 3, and the number of GNN layers to 2.

A.4 Performance on Flu dataset

The results on the Flu dataset are detailed in table 2. By applying our method, we can synergistically combine these sources to enhance prediction accuracy. Our approach has consistently outperformed other methods by at least 9% across all metrics, demonstrating its effectiveness in integrating and leveraging multi-source data for superior predictive outcomes.

A.5 Efficiency analysis

Here we compare the average training and testing runtime per epoch. For statistical methods that do not have epoch training,

Table 2: The performance on flu dataset.

Dataset	Method	MAE	RMSE	EVS	CoD	Pearson
Flu	SRA-MLP	0.062	0.095	0.115	0.107	0.351
	RR-XGBoost	0.744	0.859	0.168	0.168	0.440
	NARGP	0.045	0.066	0.379	0.376	0.620
	DMSP	**0.038**	**0.060**	**0.536**	**0.536**	**0.733**

we record the optimization time for one iteration. Some methods need to do data pre-processing before they can make predictions, and this procedure can be saved into the dataset. To make a fair comparison, we only record the time used for forward and backward operations for the same number of predictions. As shown in Table 3, the training time of our proposed method is between the two machine learning models and is smaller than NARGP by one order of magnitude. For the testing phase, our model has a slightly higher runtime than the NARGP method.

Table 3: Running time for Training and Testing

Methods	SRA-MLP	RR-XGBoost	NARGP	DMSP
Training	0.0325s	0.0963s	0.5732s	0.0685s
Testing	0.0011s	0.0028s	0.0163s	0.0201s