RESEARCH ARTICLE

Performance and Generalizability Impacts of Incorporating Geolocation into Deep Learning for Dynamic $PM_{2.5}$ Estimation

Morteza Karimzadeh[a], Zhongying Wang[a], and James L. Crooks[b]

[a]University of Colorado Boulder; [b]National Jewish Health

ARTICLE HISTORY
Compiled October 22, 2025

ABSTRACT
Deep learning models have demonstrated success in geospatial applications, yet quantifying the role of geolocation information in enhancing model performance and geographic generalizability remains underexplored. A new generation of location encoders have emerged with the goal of capturing attributes present at any given location for downstream use in predictive modeling. Being a nascent area of research, their evaluation has remained largely limited to static tasks such as species distributions or average temperature mapping. In this paper, we discuss and quantify the impact of incorporating geolocation into deep learning for a real-world application domain that is characteristically dynamic (with fast temporal change) and spatially heterogeneous at high resolutions: estimating surface-level daily $PM_{2.5}$ levels using remotely sensed and ground-level data. We build on a recently published deep learning-based $PM_{2.5}$ estimation model that achieves state-of-the-art performance on data observed in the contiguous United States. We examine three approaches for incorporating geolocation: excluding geolocation as a baseline, using raw geographic coordinates, and leveraging pretrained location encoders. We evaluate each approach under within-region (WR) and out-of-region (OoR) evaluation scenarios. Aggregate performance metrics indicate that while naïve incorporation of raw geographic coordinates improves within-region performance by retaining the interpolative value of geographic location, it can hinder generalizability across regions. In contrast, pretrained location encoders like GeoCLIP enhance predictive performance and geographic generalizability for both WR and OoR scenarios. However, our qualitative analysis reveals artifact patterns caused by high-degree basis functions and sparse upstream samples in certain areas, and our ablation results indicate varying performance among location encoders such as SatCLIP vs. GeoCLIP. To the best of our knowledge, this is a first integration and systematic evaluation of location encoders in a complex, temporally dynamic estimation scenario. In addition to guiding better model development for air pollution estimation and location encoders, this study provides insights for effective incorporation of location into deep learning for geospatial predictive tasks.

KEYWORDS
geolocation; location encoder; air pollution; deep learning; generalizability;

1. Introduction

Location plays a central role in spatial methods, including quantitative spatial analysis and geographic information science. The first law of geography by Waldo Tobler, while rooted in empirical evidence, formalizes this importance based on spatial dependence

CONTACT Morteza Karimzadeh. Email: karimzadeh@colorado.edu

in rather simple terms: "Everything is related to everything else, but near things are more related than distant things" [1]. Inferential statistical methods, such as linear regression models, which are originally designed for independent samples, need to be modified for application to spatial data. Modifications are well studied and formalized in spatial statistics models, such as spatial lag or spatial error models [2], or geostatistical interpolation methods such as Kriging [3]. The list of methods for which location takes a central role is not short, and examples in this paragraph are just pointers to some of the more commonly-recognized methods.

The same principles of spatial dependence, connectivity, and location underlie deep learning (DL) architectures, which have revolutionized many domains in science and now in society under the umbrella brand of artificial intelligence (AI). In fact, the most successful and revolutionary architectures of deep learning are those that successfully leverage spatial context (i.e., spatial dependence) [4,5], or temporal context [6], or both [7]. These deep learning architectures have found significant performance advantages over context-agnostic prediction models. In context-agnostic prediction, observations of x_i at location i are used to predict y_i at the *same* location. Instead, modern deep learning models leverage observations made at spatial and or spatiotemporal context neighborhood to enhance estimations.

While advances in deep learning excel at leveraging *spatial* information, optimal ways of leveraging *geographic location information* remain underexplored. Note that several methods are proposed to *transform* geographic (or projected) coordinates into features for deep learning [8], however, regardless of (and beyond) the transformations used, the general higher-level approach in incorporating location into deep learning, which is still an active area of research, is our focus here.

It is crucial that we first differentiate between spatial and geographic. In the context of a Convolutional Neural Network (CNN), for example, the network learns to identify patterns within the image coordinates system, for instance around point (x, y) in image coordinates, without developing an understanding of geographic location s_i corresponding to latitude φ and longitude λ mapped to (x, y). However, how to *best* incorporate location (i.e., latitude φ and longitude λ) to maximize generalizability remains an open question, with little research quantifying the generalizability impacts of doing so using different methods on temporally-dynamic estimation scenarios. This is partly due to the gap between studies applying deep learning in geospatial sciences and environmental monitoring, where predictive performance is measured on training and testing sets with minimal interrogation of the *black box* nature of deep learning, and the scarcity of fundamental studies *quantifying the behavior* of (geo)location information within deep learning frameworks. Investigating the role of location in deep learning-based estimation is even more important given the emergence of ambitious *location encoders* [9–11], which attempt to summarize all relevant information at location s into an embedding vector e_s, suitable for ingestion by deep learning algorithms, for improved predictive performance, and more importantly, geographic generalizability.

Prior research has made significant progress in identifying strategies to incorporate spatial dependence [12] or geographic information into deep learning, including direct use of raw coordinates, hand-engineered spatial features, and learned location embeddings. Mai et al. [8] provide a comprehensive survey of coordinate transformation methods—such as sinusoidal encodings, spatial tile embeddings, and kernel-based representations. Separately, contrastive learning frameworks such as Contrastive Spatial Pre-training (CSP) [9], SatCLIP [10], and GeoCLIP [11] have adapted CLIP-style Contrastive Language-Image Pre-trainig architectures [13] to align image and loca-

tion representations in a shared latent space (dubbed Contrastive Location-Image Pre-training), enabling pretrained models to distill contextual geographic knowledge. However, evaluations of these location encoders have been limited to static tasks such as species distribution or primitive tasks such as predicting long-term climate averages, where temporal dynamics and real-time variability are less critical, and real-world application is limited. On the other hand, studies such as [14–16] have proposed robust spatial validation—such as checkerboard partitioning or distance-aware cross-validation—to more accurately assess geographic generalizability. Yet, the interplay between location encoding strategies and spatial evaluation methodology remains underexplored, particularly for dynamic, high-resolution prediction tasks.

In this paper, we take a step towards closing this gap by quantifying and analyzing the impact of incorporating geolocation information of locations s_i with latitude φ and longitude λ in deep learning within a real-world temporally-dynamic application context of estimating daily surface-level $PM_{2.5}$ from remote sensing imagery. We first offer an in-depth discussion of different approaches to incorporating geolocation into deep learning, including simple featurization to leveraging the state-of-the-art location encoders. We then characterize the domain of surface-level air pollution estimation, including a brief overview and justification for selecting this application domain as an appropriate test-bed for this study. We then present a series of experiments and results from two complementary aspects of predictive (i.e., estimation) performance and geographic generalizability using Within-Region (WR) and Out-of-Region (OoR) evaluation scenarios. We present an ablation study to compare alternatives, and complement the experimental results with qualitative analysis of estimations in the contiguous United States. We characterize the findings and suggest directions for future research.

While the primary contribution of this paper is centered on methods of incorporating geolocation into deep learning, our work also contributes to the literature on surface-level air pollution estimation. Recent advances in surface-level $PM_{2.5}$ estimation increasingly leverage deep learning architectures that capture spatiotemporal contextual observations, as reviewed in depth by a recent survey [17]. Architectures combining convolutional and recurrent layers, such as CNN-LSTM models [18], have shown improved accuracy by jointly modeling spatial features and temporal dynamics. Graph-based models, including Graph Neural Networks (GNNs) and spatiotemporal Graph Convolutional Networks (GCNs), which represent spatial relationships as graphs, have been used primarily in the context of short-term forecasting at monitoring stations rather than full-coverage spatial surface level estimation [19,20]. Additionally, attention-based models capture directional flow and long-range dependencies, further improving estimation or forecasting performance [21,22]. These research efforts show a collective movement and growing emphasis on learning both spatial and temporal context to enhance the robustness and generalizability of $PM_{2.5}$ estimation frameworks.

Unlike existing $PM_{2.5}$ estimation established products that rely on context-agnostic regression with raw geographic coordinates [23], treat spatial context as *static* features like land use or proximity to emission sources [24], or more recent research that leverage spatiotemporal *context* [17], our current study advances the field by explicitly interrogating the role of geolocation features in deep learning for dynamic, high-resolution air pollution estimation. Our prior work introduced a state-of-the-art Bi-LSTM with attention architecture that demonstrated superior performance, particularly during high-pollution events such as wildfires, by incorporating temporal inputs, including aerosol and meteorological data, and by integrating wildfire smoke density as a pre-

dictive covariate [25–27]. This paper extends that study and evaluates and compares multiple geolocation integration strategies within the modeling framework. By evaluating both within-region (WR) and out-of-region (OoR) performance using several spatial partitioning schemes, we move beyond conventional random more simple spatiotemporal validation schemes common in air pollution studies. This explicit attention to geographic generalizability and the evaluation of location encoder representations in a dynamic prediction task distinguishes our work from prior efforts, including Di et al. [23], Wei et al. [28], and even our own earlier study [25], which did not examine location feature representations or spatial generalization as central research questions.

To the best of our knowledge, this study is the first quantification of the impact of geolocation features and systematic evaluation of location encoders in a complex, temporally-dynamic estimation scenario, thereby expanding the empirical basis for how geolocation should be integrated into geospatial deep learning.

2. Materials and Methods

2.1. Formalizing Spatiotemporal Context

To formalize spatial and temporal context in this manuscript, consider the following:

2.1.1. Spatial Context

Given a set of spatial locations $S = \{s_1, s_2, \ldots, s_n\}$, where s_i represents the coordinates of location i, the observation x_i at location s_i is influenced by its spatial neighborhood $N_s(s_i) = \{s_j : d(s_i, s_j) \leq r, j \neq i\}$, where $d(\cdot, \cdot)$ can be a distance metric and r is a threshold radius (or another definition of connectivity or neighborhood). A spatial prediction model can be formulated as:

$$y_i = f(x_i, \{x_j : j \in N_s(s_i)\})$$

where $f(\cdot)$ is the prediction function learned by the model. Among such models, Convolutional Neural Networks (CNNs) [29] are notable. CNNs operate on data arranged on a regular spatial grid, where features are extracted from each pixel in relation to its local neighborhood defined by a convolution window. This architecture has been extensively used in geospatial and remote sensing applications, spanning tasks in both land [30] and ocean remote sensing [31].

2.1.2. Temporal Context

For temporal sequences, consider a discrete time index $T = \{t_1, t_2, \ldots, t_T\}$. The observation x_i^t at location i and time t is influenced by its past observations $\{x_i^{t-k}, \ldots, x_i^{t-1}\}$ for some lag k. A temporal prediction model is represented as:

$$y_i^t = f(x_i^t, x_i^{t-1}, \ldots, x_i^{t-k})$$

Recurrent Neural Networks (RNNs) [32] and Long Short-term Memory networks (LSTMs) [6] belong to this group, and have been successfully adopted in geospatial applications for time-series forecasting, such forecasting the geographic spread of infectious diseases [12] or various remote sensing applications [33]. Temporal models such as this actually allow implicit incorporation of absolute location s_i (rather

than relative location as in spatial context). For example, in geographic disease forecasting, leveraging a temporal model for each spatial unit i inherently captures the disease dynamics at location s_i to improve forecasting of y_i^t. If this implicit capturing of (geo)location is to be made explicit, then geolocation can also act as a predictive feature such that, $y_i^t = f(x_i^t, x_i^{t-1}, \ldots, x_i^{t-k}, s_i)$, where s_i encodes geographic location information such as latitude and longitude, spatial unit ID, or pre-learned embeddings (i.e., encodings), as discussed later in this paper.

2.1.3. Spatiotemporal Context

Combining both spatial and temporal dimensions, the prediction at location i and time t depends on the spatiotemporal neighborhood:

$$y_i^t = f(x_i^t, \{x_j^{t-l} : j \in N_s(s_i), l = 0, 1, \ldots, k\})$$

In this family of models, ConvLSTMs [34] capture spatiotemporal context and have gained adoption in environmental monitoring [35]; Transformers [4] are being employed for remote sensing and geospatial applications [36], and spatiotemporal graph neural networks have shown promise in spatial epidemiology [16], or even in remote sensing for building change detection [37]. The taxonomy above provides a non-exclusive overview of deep learning approaches incorporated in geospatial data science; for instance, sequence-to-sequence methods are not discussed here, but the reader can refer to [38].

The intention is to highlight that the current revolutionary impact of deep learning methods is in large due to their ability to successfully capture spatiotemporal *context*, and how there is potential for further enhancement if (geo)location information is preserved within the spatiotemporal context.

2.2. Geolocation and Deep Learning

Deep learning is well capable of capturing spatial and temporal *context*; however, methods to incorporate explicit *geographic location* remains an active area of research. In this section, we examine approaches to integrating geolocation—specifically latitude φ and longitude λ—into deep learning for geospatial estimation tasks. For clarity, we focus on the role of location s_i in augmenting a set of observed predictive variables x_1 through x_m, with the goal of improving both estimation accuracy and geographic generalizability. The methods discussed here can be extended to temporal and spatiotemporal context models, including the framework used in our experiments.

2.2.1. Approach 1: No Geolocation Feature

If no geolocation information is furnished to the model, then the model must learn a function $f : \mathbb{R}^m \to \mathbb{R}$ that maps the input observations $x_1, x_2, \ldots, x_m$ directly to the target variable y, such that:

$$y = f(x_1, x_2, \ldots, x_m)$$

where $f(\cdot)$ is learned through optimization during model training. One may ask what the purpose of intentionally leaving out the geolocation information might be: If the objective is to learn the global (albeit non-linear) mapping of observations to the target

irrespective of geographic location, then excluding geographic location is justified, ensuring potentially better geographic generalizability to observations made in regions unseen by the model during training, because geolocation is not used as a predictive variable after all. Leaving out location forces the model to learn the complex and non-linear relationships between the predictors and the target, rather than using location as a predictive feature.

2.2.2. Approach 2: Geolocation as Geographic Coordinates Features

This naïve approach involves directly including latitude φ and longitude λ as input features in the model. Given that geographic coordinates are cyclical in nature and measured on a spherical surface, a direct input of φ and λ might mislead the model in regions of transition. To handle this, a common practical technique is to transform coordinates using sine and cosine functions:

$$\text{lat}_s = \sin(\varphi), \quad \text{lat}_c = \cos(\varphi), \quad \text{lon}_s = \sin(\lambda), \quad \text{lon}_c = \cos(\lambda)$$

Using this transformation technique, coordinates close to each other geographically remain close in the transformed feature space, making it easier for the model to learn the geospatial relationship of observations and geolocation. The model then learns a function $f : \mathbb{R}^{m+4} \to \mathbb{R}$ that maps the observations $x_1, x_2, \ldots, x_m$ along with $\text{lat}_s, \text{lat}_c, \text{lon}_s, \text{lon}_c$ to the target y:

$$y = f(x_1, x_2, \ldots, x_m, \text{lat}_s, \text{lat}_c, \text{lon}_s, \text{lon}_c)$$

While a variety of coordinate transformation strategies have been surveyed before [8], we adopt the sinusoidal encoding as a representative example to evaluate the impact of geolocation features. In our contiguous U.S. study region, where latitude and longitude vary smoothly without discontinuity, the choice of transformation method has limited impact on generalizability.

Including φ and λ as input features enables deep learning models to learn complex, location-dependent mappings to the target variable y. While this can improve performance in training regions, it also risks overfitting to location-specific patterns. For example, the winning team of the AutoICE challenge [39] reportedly [40] used geolocation features to enhance performance in a test region that overlapped with training data (the test images in that competition were situated within the spatial extent of the training data). Put differently, geographic overlap can inflate apparent generalization when location is used as a feature.

On the other hand, incorporating φ and λ as input features can be problematic if the predictive observations $x_1, x_2, \ldots, x_m$ are weak predictors of the target y, resulting in the model to overly rely on the geographic coordinates to differentiate between observations, leading to overfitting on location-specific patterns rather than the actual underlying predictive relationship.

Furthermore, using geographic coordinates can potentially harm model generalizability to other regions, even when observations are predictive of the target. Geolocation may act as a proxy for missing predictors or unmeasured covariates (similar to spatial statistical models), capturing region-specific influences that are not explicitly included as features. However, when the model is deployed in regions not represented in the training data, the absence of those region-specific patterns can lead to degraded performance, as the model's learned mapping relies heavily on location-based proxies

that do not transfer well to unseen areas.

To prevent overfitting to observations in a specific region, one potential solution is to use cross-validation or early stopping to balance overfitting and underfitting, a topic that has been studied by recent research for geospatial data [15]. However, these studies have not addressed the impacts of the use of geolocation as features in the model, and instead, focused on the broader issues of geographic generalization, even if only direct observations are involved.

2.2.3. Approach 3: Geolocation as Pre-learned Location Encodings

An alternative is to use **pretrained** location encoders to generate embeddings e_s for each location s and ingest those embeddings in the predictive task at hand (i.e., the downstream task). These embeddings are essentially d dimensional vectors, akin to applying dimensionality reduction to all imaginable data at a given location (in an idealistic scenario) and reducing that data to the d dimensions of the embedding vector.

Location encoders aim to capture the attributes of any given location in a latent embedding vector. To better understand what these location encoders are, we must first quickly cover their inspiration: the CLIP (Contrastive Language-Image pretraining) framework [13], initially developed by OpenAI for aligning embeddings of natural images and their captions. CLIP uses contrastive learning, training on large batches of image-caption paired data, where during (pre)training using a self-supervised approach, embeddings of positive pairs (image-caption pairs) are pulled closer together, while embeddings of negative pairs (non-matching images and captions) are pushed apart. Trained on vast amounts of naturally occurring image-caption pairs collected from the internet, CLIP results in embedding vectors for positive pairs that are *aligned*. Put differently, if an image is fed to the image encoder of CLIP, and the corresponding caption is fed to the text encoder of the CLIP, they both result in embedding vectors that are very close to each other in terms of cosine similarity, and therefore, aligned in the high dimensional embedding space. CLIP-based image encoders have demonstrated superior performance over fully-supervised learning methods on benchmark datasets [13].

Location encoders keep the image encoder in CLIP as the component of the network extracting features from (satellite or street-view) imagery, but replace the language (text) encoder with a *location encoder*. This location encoder is a learnable function that operates on a high-dimensional representation of geographic coordinates produced by a fixed *positional encoder* ϕ. The positional encoder is often based on Fourier features (e.g., in GeoCLIP) or spherical harmonics (e.g., in SatCLIP), and its goal is to expand latitude φ and longitude λ into a higher dimension feature space that captures spatial patterns at different scales.

A general formulation of the positional encoder is:

$$\begin{aligned}\phi(\varphi,\lambda) = [\ & \sin(2^0\pi\varphi), \cos(2^0\pi\varphi), \ldots, \sin(2^{k-1}\pi\varphi), \cos(2^{k-1}\pi\varphi), \\ & \sin(2^0\pi\lambda), \cos(2^0\pi\lambda), \ldots, \sin(2^{k-1}\pi\lambda), \cos(2^{k-1}\pi\lambda)]\end{aligned}$$

where k is the number of frequency bands used in the expansion. The location encoder is then defined as a learnable neural network $g : \mathbb{R}^{2k} \to \mathbb{R}^d$ that transforms this fixed ϕ representation into the final embedding vector e_s:

$$e_s = g(\phi(\varphi, \lambda))$$

During pretraining, this embedding e_s is aligned with the embedding of an image taken at location (φ, λ), using a contrastive loss. In other words, the image encoder distills geographic attributes (such as terrain cover, built environment, vegetation patterns) into the location encoder, effectively storing visual and contextual cues in the location embedding. In CSP [9], g is trained using ground-level imagery focused on species distribution and natural environments; in GeoCLIP [11], Flickr images; and in SatCLIP [10], Sentinel-2 imagery.

At inference time for downstream tasks, the pretrained location encoder receives raw latitude and longitude coordinates and returns a static embedding $e_s = g(\phi(\varphi, \lambda))$. This vector encodes location-specific attributes learned during pretraining and remains fixed during downstream model training. The downstream model can then integrate e_s alongside other predictive inputs (e.g., remote sensing or meteorological data), allowing it to leverage implicit spatial context—such as land use, climate zone, or infrastructure—without manually engineering or deriving these variables.

This two-stage formulation—first mapping (φ, λ) into a high-frequency basis via ϕ, then learning g to align the result with image features—is foundational to modern CLIP-style location encoders. It enables geographic generalization by decoupling geographic position from learned visual cues, and supports plug-and-play use in various downstream geospatial prediction tasks.

The use of multi-frequency trigonometric functions in $\phi(\varphi, \lambda)$ allows the model to represent spatial variation at multiple scales, a key feature for learning from geospatial data with processes at different spatial resolutions. For example, consider $PM_{2.5}$ concentrations: large-scale atmospheric transport can cause regional haze events spanning hundreds of kilometers (e.g., wildfire smoke spreading across states), while at the same time, local emissions from urban centers or industrial zones create fine-scale variability within tens of kilometers.

In the Fourier-based positional encoder, low-frequency terms (e.g., $\sin(2^0\pi\varphi)$, $\cos(2^0\pi\lambda)$) represent coarse-scale patterns that vary slowly over space, enabling the model to recognize broad spatial gradients like continental east-west transport. In contrast, higher-frequency terms (e.g., $\sin(2^{k-1}\pi\varphi)$) allow the model to resolve fine-scale features, such as abrupt $PM_{2.5}$ spikes near city centers or downwind of industrial corridors.

The long-term vision of location encoders is to allow downstream models to benefit from location-specific knowledge without the need to collect and include all observations for every predictive task. This, theoretically should reduce the risk of overfitting, as the model is not directly *memorizing* the training data but rather, learning a generalized representation of geographic context across tasks and regions. While this vision is grand, location encoders are a recent line of research, and their performance so far is evaluated on simple tasks such as average yearly temperature or other static targets, given that location encoders generate a *static* vector representing locations irrespective of season or day, or any real-time observation, for that matter. In this manuscript, we focus on leveraging location encoders and their impact on generalizability and performance for estimating a dynamic, ever-changing target: daily estimation of air pollution component, $PM_{2.5}$.

2.3. Experiments

2.3.1. Domain Characterization: Estimating Surface-level $PM_{2.5}$

Estimating surface-level air pollution, and in particular, $PM_{2.5}$ is important to quantify for public health studies. Prolonged exposure to high concentrations of $PM_{2.5}$ has been associated with respiratory and cardiovascular diseases, and premature mortality [41]. Accurate, high-resolution spatiotemporal $PM_{2.5}$ estimates have several important uses in policy-making and public health. Estimates covering recent years are used to quantify $PM_{2.5}$ human exposure in epidemiological or clinical cohort studies evaluating the impact of air pollution on specific health outcomes, which in the U.S. feed directly into regulatory decisions under the Clean Air Act. Predictions covering future days, on the other hand, are used for public health warnings, while predictions encompassing future years are used to evaluate the effects of regulatory or climatic changes.

In the United States, the Environmental Protection Agency (EPA) has deployed a series of carefully-calibrated Air Quality System (AQS) monitors to measure the spatially and temporally variable concentrations of $PM_{2.5}$. However, public health studies require estimation of surface-level pollution at the addresses of study cohort members, while the AQS stations may be far away and spatially-sparse. On the other hand, satellite observations can be used to estimate particulates suspended in *columns* of air, and therefore, the predictive task is to use satellite observations and ancillary data to derive surface-level (rather than total column) concentrations of $PM_{2.5}$. The machine learning *target* is the surface-level $PM_{2.5}$ levels measured by AQS sensors. Satellite observations are often leveraged for this purpose. Most notably, Aerosol Optical Depth (AOD), which quantifies the extinction of solar radiation by aerosol particles integrated throughout the atmospheric column, provides valuable large-scale coverage. However, it does not directly measure surface-level concentrations and exhibits a complex, nonlinear relationship with $PM_{2.5}$ due to meteorological influences, aerosol type, and vertical distribution.

$PM_{2.5}$ is spatially and temporally variable, making remote sensing and deep learning approaches vital for generating high-resolution, daily estimates that AQS monitoring stations alone do not provide. Integrating geolocation into deep learning models has the potential to further enhance these estimates by capturing location-specific pollution sources, meteorological influences, and landuse patterns, and therefore, we believe this estimation task to be an appropriate test bed for examining the impacts of different ways of incorporating geolocation into deep learning.

We specifically focus on the task of estimating *daily* concentrations, which can change from day to day, to investigate the impact of leveraging geolocation information or static location encodings, given that at the time of this writing, no spatio*temporal* location encoding has been developed yet, to our knowledge. To keep our experiments further grounded, we leverage an existing and proven model with state-of-the-art performance for estimating surface-level $PM_{2.5}$ in the continental United States [25].

2.3.2. Model Architecture

Our base model for $PM_{2.5}$ concentration prediction (Fig.1(a)) follows a Bidirectional Long Short-Term Memory (Bi-LSTM) [6,42,43] network with Luong Attention [44], as detailed in [25]. In brief, a 21-day window of time-series multi-variate input is fed into a Bi-LSTM to capture forward and backward dependencies in $PM_{2.5}$-related features. This 21-day window allows capturing of potential weeks-long persistence of $PM_{2.5}$ episodes which previous research has identified [45], giving the model ample tempo-

ral context while keeping computational demands manageable. Exploratory modeling confirmed that this length achieved the lowest error, with longer windows providing no additional improvement.

Inputs to the model include satellite-derived Aerosol Optical Depth (AOD) [46], meteorological variables (e.g., temperature, precipitation, wind direction/speed) [47,48], wildfire smoke density [26,27], elevation [49], Normalized Difference Vegetation Index (NDVI) [50], K-Nearest Neighbors Inverse Distance Weighted (IDW) $PM_{2.5}$ measurements to incorporate ground-based spatial context, as well as temporal encodings (sine and cosine of (Day of the Year, Month of the Year), and Year). Each predictor feature was re-projected to a common 1 km grid (MODIS Sinusoidal) and, when necessary, resampled to match. Daily variables were matched by calendar date, while coarser products—such as the 16-day MODIS NDVI—were assigned to each estimation day using the nearest-available composite. The input datasets (predictors) and target are summarized in Table 1.

Table 1. Datasets, features (predictors), and target used in this study.

Category	Variables / Features	Source (Product)	Spatial Res.	Temporal Res.	#
Target	Surface-level $PM_{2.5}$ concentration	U.S. EPA AQS	Station (point)	Daily	1
Predictors	Aerosol Optical Depth (Blue 0.47 μm; Green 0.55 μm)	MODIS MCD19A2.061 (MAIAC)	1 km	Daily	2
	Meteorology: dayl, prcp, srad, tmax, tmin, vp	Daymet	1 km	Daily	6
	Meteorology: wind direction (th), wind speed (vs)	gridMET (1/24°)	∼4 km	Daily	2
	Wildfire Smoke Density (WSD)	NOAA HMS Smoke	Polygon	Daily	1
	Elevation	GMTED2010	1 km	Static	1
	NDVI (combined)	MODIS MCD43A4	500 m	16-day	1
	KNN–IDW $PM_{2.5}$ (from neighboring stations)†	Derived from AQS (9-NN)	N/A	Daily	1
	Temporal encodings: sin / cos (DOY, Month), Year	Derived	N/A	Daily	5
Geolocation (variants)	Latitude, Longitude	Derived (coordinates)	N/A	Static	2
	sin / cos of lat, lon	Derived (coordinates)	N/A	Static	4
	GeoCLIP location embedding	Pretrained location encoder	N/A	Static	512-D

Notes: Reprojection/temporal matching as described in Sec 2.3.2; nearest-available NDVI composite is used for each estimation day. †IDW excludes a station's own measurement and uses only its 9 nearest neighbors to avoid data leakage.

These features are selected to capture the primary drivers of surface-level $PM_{2.5}$ concentrations. AOD is satellite-observed columnar aerosol loadings, which, though not a direct measure of surface-level concentrations, is predictive of surface-level pol-

lution under certain atmospheric conditions. Meteorological variables influence pollutant dispersion, chemical transformation, and accumulation. Wildfire smoke density directly relates to episodic spikes in $PM_{2.5}$ primarily due to biomass combustion. NDVI provides information on vegetative cover, which can modulate emissions and pollutant deposition. Elevation influences local meteorology and pollutant trapping, while the KNN-IDW interpolated $PM_{2.5}$ ~~layer~~ input feature reflects observed surface-level spatial trends from monitoring stations. The supervised target in this study is the observed daily PM2.5 concentration *measured at AQS monitoring stations*; the KNN–IDW $PM_{2.5}$ is used only as an auxiliary predictor to convey local spatial context, not as a target or intermediate surface. Note that the IDW-interpolated $PM_{2.5}$ excludes self-measurements of a monitoring station during training or evaluation, and instead, uses only the 9 nearest neighboring stations' ground-level measurements to calculate the interpolated values. This is to ensure there is no leakage in training data, and that estimates can be made for any point on the surface. The interpolated feature represents only a distance-weighted neighborhood *average value* and does not encode the spatial identity or exact location of any monitoring station. Because the Bi-LSTM model has no spatial receptive field and processes each sample independently, it cannot memorize or infer the true values at held-out test stations. Also, note that because inverse-distance weighting rapidly decreases the influence of distant stations, this feature reflects primarily the immediate spatial context rather than a continuous interpolated surface. The KNN–IDW feature may be less informative in areas with sparse monitoring; however, the Bi-LSTM model uses it jointly with other predictors—such as AOD, meteorology, and vegetation—that more directly capture the pollution level in those regions. ~~Temporal encodings help capture seasonal patterns in emissions.~~

Because AOD is a key predictor of surface $PM_{2.5}$ and a time-series of it is fed to the model for every pixel, missingness caused by cloud or snow cover were first imputed with a lightweight random forest regressor trained on the same set of predictors (excluding the KNN-IDW $PM_{2.5}$ and the target itself), as detailed in [51]. After this step, a single masking layer flags any residual missing entries across all predictors, enabling the Bi-LSTM to downweight them during training rather than treat them as observed values [52].

Layer normalization and dropout [53] are applied after each Bi-LSTM layer to stabilize training and mitigate overfitting. The Luong attention mechanism [44] then learns to focus on the most informative timesteps within the 21-day window by computing a context vector as a weighted average of the hidden states. The representation is then passed through a fully connected layer to make the final prediction for $PM_{2.5}$. We refer the reader to [25] for detailed information on data processing, implementation, long-term evaluation of the model and comparison to multiple other baselines and benchmarks. What is different in this manuscript is the ways in which geolocation features are fed to the model, and the way performance is evaluated under regional conditions. Specifically, we use several configurations in regional partitioning for creating training and testing splits as detailed in the Section 3, and for each configuration, train the model with four general variations—differing in the way geolocation is incorporated:

(1) **No geolocation input**: The model does not receive geolocation (φ, λ) information, similar to Approach 1 described in section 2.2, forcing the model to purely rely on time-series observations for estimating the target.
(2) **Direct geolocation input**: We append the raw latitude and longitude values (φ, λ) to the time-series feature vector at each timestep, similar to Approach

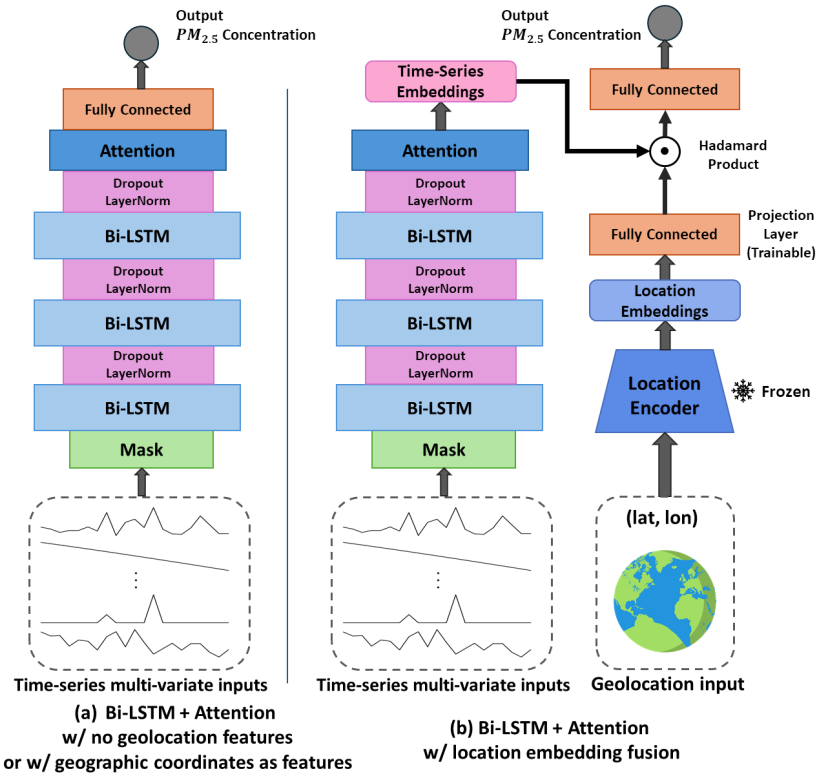


Figure 1. Model architecture for estimating surface-level $PM_{2.5}$ concentrations enhanced with a) no geolocation info, or geolocation appended as geographic coordinates features, and b) geolocation embeddings vector extracted from the pretrained location encoder and fused through the Hadamard product operation.

2 described in section 2.2, allowing the model to rely on both the time series observations as well as geolocation for estimating the target.

(3) **Sinusoidal transformation of geographic coordinates**: Also similar to Approach 2 in section 2.2, but with a transformation applied first: instead of passing latitude and longitude values directly in angular units, we apply sinusoidal transformations to preserve their cyclical properties:

$$\text{lat}_s = \sin(\varphi), \quad \text{lat}_c = \cos(\varphi), \quad \text{lon}_s = \sin(\lambda), \quad \text{lon}_c = \cos(\lambda)$$

These four transformed features are then appended to the time-series inputs, allowing the model to rely on both the time series observations as well as transformed geolocation for estimating the target.

(4) **Fusing location encoder embeddings**: We integrate **pre-learned** location embeddings from the pretrained location encoder GeoCLIP [11]. GeoCLIP is a location encoder trained using a CLIP-style contrastive learning framework to align geotagged Flickr images with geographic coordinates (section 2.2.3 above covers details of this approach). Its image encoder is a Vision Transformer (ViT-B/16), while the positional encoder is a Fourier feature mapping of latitude and longitude, capturing multi-scale spatial patterns. During pretraining by the original developers [11], GeoCLIP was trained on over 1.2 million image-location pairs globally sampled from the YFCC100M dataset, with relatively denser coverage in urbanized and photo-rich regions, which is important for $PM_{2.5}$ estimation in

areas more prone to generate emissions. We will contextualize GeoCLIP against other available location encoders in section 3.3.

As shown in Fig.1(b), the location encoder outputs a vector e_s, summarizing geographic information for a given latitude and longitude. In line with the original intent of location encoders, we freeze the location encoder remains during training to preserve the learned spatial representations. A trainable projection layer is applied to e_s to align its dimensionality with the time-series embeddings e_{ts} produced by the Bi-LSTM with Attention branch. Specifically, let $e_{proj} = \text{Projection}(e_s)$ denote the projected location embeddings. We then fuse e_{proj} and e_{ts} via a Hadamard product:

$$e_{fused} = e_{ts} \odot e_{proj}$$

where $\odot$ denotes Hadamard product. This operation allows the model to learn the interactions between temporal patterns and geographic context. The fused embedding e_{fused} is then passed to a final fully connected layer that regresses the surface-level $PM_{2.5}$ concentrations, allowing the model to rely on both the time-series observations as well as pre-learned embeddings for the given location.

2.3.3. Model Training

The training logistics settings remain the same as the original model described in [25]. Our region, similarly, encompasses the Continental United States (CONUS), but we limit our daily estimation to all days in 2021 to keep computations manageable for this paper. Inputs are normalized feature-wise to the range of [-1,1] using a MinMax Scaler. We employ the Adam optimizer [54] with an exponential learning rate scheduler [55] (initial learning rate of $1e^{-3}$, decay factor of 0.8 every 30,000 steps). All models are trained for 100 epochs with a batch size of 256, and Huber loss is used to mitigate the impact of outliers in the $PM_{2.5}$ data [56].

3. Results

We assess model performance and generalizability using three main evaluation settings, followed by ablation and qualitative analyses. In all experiments, we report the coefficient of determination (R^2), root mean square error (RMSE), and mean bias error (MBE) as performance metrics for average of five training runs. Complete evaluation tables are provided in Appendix Table A1-A4. We provide key metrics in the following Sections. Please note that the results **within** each setting can be compared, not across settings, to ensure the number of training samples, spatial distributions, and all other configurations remain similar.

3.1. Within-Region (WR) Evaluation

Within-region scenarios are ones where the test locations are geospatially in-distribution in comparison to training locations. In other words, these experiments are designed to see if incorporating geolocation information enhances the model's predictive performance and adds geospatial *interpolative value*.

Random Test Set Evaluation: In this setting, 10% of the samples are randomly selected as the test set. Because the model uses multivariate time-series inputs, test

samples may include locations whose temporal observations (at different timesteps) were seen during training.

Spatial Test Set Evaluation: To evaluate the model's ability to predict at unseen locations, we create a spatial test set by randomly dropping 10% of the unique locations from the dataset. While this approach tests spatial generalizability and is common for air pollution estimation evaluation [41], it is worth noting that the random selection process does not enforce spatial clustering—meaning that the training set may still include locations that are geographically close to test locations. Put differently, this experiment still tests for Within-Region (WG) performance.

(a) Random test set metrics

(b) Spatial test set metrics

Figure 2. Within-region model performance under Randomly (a) and Spatially (b) assigned test sets.

Figure 2 shows performance metrics for Within-Region scenarios. The results of the first three columns are not entirely surprising: adding geolocation information to the model while training enhances its ability to generalize to test locations that are close

to or are within the training regions. Sinusoidal wrapping of geographic coordinates helps compared to a simple direct incorporation of degree-measurements of geographic coordinates, even though the region of the study is contained within the continental United States, where latitudes range from approximately 24°N to 49°N and longitudes range from 125°W to 67°W without discontinuities (such as those at poles or the International Date Line).

What is more interesting is that GeoCLIP-extracted location embeddings help improve performance even further. It is worth remembering the context: our predictive features do include multi-variate observations of AOD measurements and other ancillary variables, and the GeoCLIP-extracted embeddings are static vectors for each location, derived partially from images on Flickr related to each location during pretraining. Although GeoCLIP embeddings are not based on satellite observations, they nevertheless help improve test performance on unseen locations for within region evaluation scenarios. The WR test results indicate that geolocation does indeed add interpolative value regardless of the selected approach to incorporate location: Both the naïve incorporation of geographic coordinates or through pre-learned location encoder embeddings, although interestingly, the prelearned embeddings increase predictive performance more than direct incorporation of geographic coordinates.

Another noteworthy observation is that, under the random test scenario, GeoCLIP with Hadamard fusion yields the smallest bias (lowest |MBE|), followed by the model without lat/lon (Table A1). Models using direct coordinate features (raw lat/lon or their sinusoidal transforms) exhibit higher bias, although none show practically significant systematic over- or under-estimation. Under the spatial split, the model without lat/lon achieves the smallest bias, while the model with GeoCLIP embeddings fusion display a slightly larger average bias than direct coordinates, but with notably smaller standard deviations across splits, indicating more stable performance across spatial partitions (Table A2).

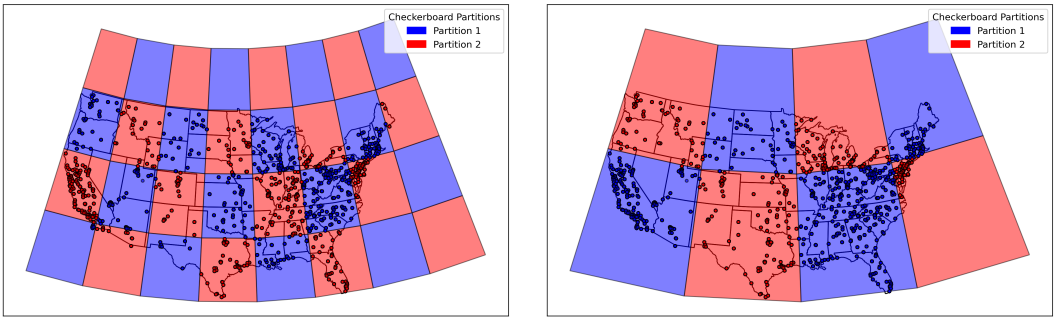
3.2. Out-of-Region (OoR) Evaluation

Geographic generalizability is broadly defined as the ability of a model trained in one region to maintain its predictive performance when deployed in another region (Our of Region - OoR), which may have distinct geographic characteristics. Robust geographic generalizability is a sought-after quality for geospatial deep learning tasks, where models are often required to make predictions in areas without sufficient training data. The ability to maintain high performance across regions enhances the model's applicability for large-scale geospatial analysis or environmental monitoring, where predictions must remain reliable despite geographic variability.

Checkerboard Partition Evaluation: For a rigorous OoR evaluation, we follow spatial partitioning designed by [14] using a checkerboard pattern, training on one partition and testing on the other set (Figure 3). This is to ensure that the training and testing sets are spatially non-overlapping and disjoint, allowing the test performance metrics to be a fair indication of model capacity to generalize to unseen regions.

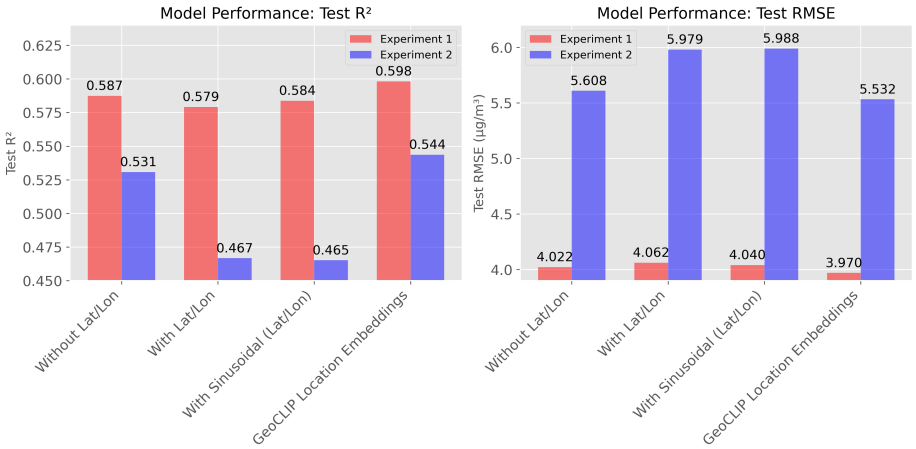
Increasing the blocks' width (denoted by δ, measured in degrees) increases the average distance between training and test observations, creating a more challenging test of the model's OoR geographic generalizability capabilities [14]. Our experiments examine two checkerboard configurations with $\delta = 8°$ and $\delta = 16°$, corresponding to moderate and more extreme spatial validation, respectively (Figure 3).

Figure 4 presents performance metrics under these settings, with the color of the

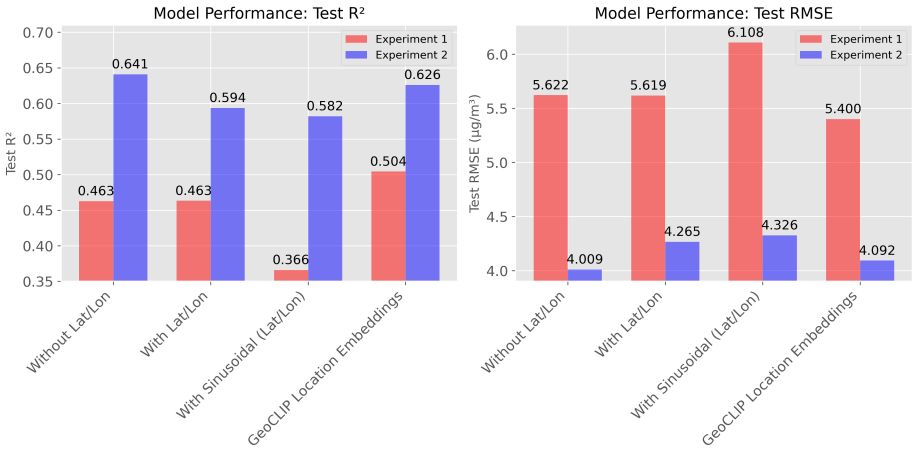


(a) Checkerboard partition with $\delta = 8.0°$ (b) Checkerboard partition with $\delta = 16.0°$

Figure 3. Checkerboard spatial partitioning used for Out-of-region (OoR) evaluations. The circles show the location of AQS stations.



(a) Checkerboard test set metrics with $\delta = 8.0°$



(b) Checkerboard test set metrics with $\delta = 16.0°$

Figure 4. Out-of-region evaluation using geospatial checkerboard partitioning with $\delta = 8.0°$ (a) and $\delta = 16.0°$ (b).

bars corresponding to the *test* partition colors on the maps. Put differently, for each value of δ, we conduct two experiment sets (with five runs for each), one with the blue partition (1) as training set and red partition (2) as test set, and another experiment with swapping these sets. This is to control for geographic variation in the location of AQS stations, and the uneven number of stations in the partitions, which is inevitable in this real-world scenario. Please also note that the results here cannot be compared to the previous section, as partitioning leads to a significant drop in the number of available training stations. Nevertheless, comparisons within each bar chart are valid (but not across the bar charts).

As can be seen in Figure 4, the models without geolocation features and the models using GeoCLIP embeddings perform the best in OoR evaluations. The models without geolocation features learn the mapping of time-series observations to daily targets, avoiding overfitting to specific regions, while GeoCLIP embeddings (in conjunction with time-series observations) appear to provide rich geographic information that allow for OoR generalization. It is worth remembering that GeoCLIP location encoders are pretrained, and even though the test regions in the checkerboard are not seen during the training of the *downstream* air pollution estimation Bi-LSTM model, those regions are *not* excluded when training GeoCLIP. Location encoders, after all, aim to distill information for every location into an embedding vector for downstream use.

The considerable drop in performance for models using naïve geographic coordinates-with or without sinusoidal wrapping-suggests that the model has learned to associate specific locations with $PM_{2.5}$ levels, rather than relying on underlying aerosol or meteorological observations. Put differently, instead of using location to disambiguate when observations are rather similar, the model overfits to spatial patterns present in the training region—reducing its ability to generalize to unseen areas.

GeoCLIP embeddings, provide the best results under the $\delta = 8.0°$ partition. However, for $\delta = 16.0°$, which present a bigger challenge to geographic generalizability, models with no geolocation feature or embedding generalize the best to the test set, with GeoCLIP-enhanced models closely following in performance. For both δ values, models with lat/lon or sinusoidal wrappings of lat/lon see a drop in performance in OoR evaluation, which is contrary to the results seen in WR evaluations. These results highlights that naïve inclusion of geolocation features and insufficient evaluation schemes that do not span WR and OoR scenarios, can hinder geographic generalizability.

Results indicate that fusing embeddings from location encoders (in this case, GeoCLIP) improves generalization even across distant regions, although the benefits somewhat diminish as the partition size increases (from $\delta = 8.0°$ to $\delta = 16.0°$). Nevertheless, GeoCLIP embeddings lead to higher performance in three out of four checkerboard scenarios for such a temporally-dynamnic estimation scenario (with observations and targets that change day to day), reinforcing the value of pretrained location encoders for better generalization by distilling complex spatial attributes, even when training and test regions are spatially disjoint. The static embeddings derived from crowdsourced imagery (e.g., in this case, Flickr) appear to encapsulate contextual information that improves predictions in geographically distant and non-overlapping regions by a model leveraging inputs of spatiotemporally-varying observations of aerosols.

Across both checkerboard partitions, especially for δ=16°, GeoCLIP with Hadamard fusion maintains the smallest bias with similar values between the two held-out folds (partitions), whereas direct coordinate features lead to larger and more asymmetric biases (Tables A3–A4). This further shows that pretrained location embeddings enhance spatial generalization, consistent with other findings.

To further assess the model's geographic generalizability under a more extreme spatial configuration, we conducted an additional experiment using a checkerboard partition with $\delta = 30°$. This partitioning effectively divides CONUS into western and eastern halves. However, this configuration presents two key caveats. First, incorporating raw or sinusoidal geographic coordinates (e.g., latitude and longitude) in such a setting is inappropriate. Specifically, the longitude values in the test region become entirely out-of-distribution (OoD) relative to the training region, undermining the utility of direct coordinate-based features (our checkerboard evaluations earlier was partially motivated to prevent this from occurring). As a result, we do not train models with naive or sinusoidal lat/lon inputs in this evaluation. Second, $PM_{2.5}$ dynamics exhibit well-known and stark differences between eastern and western CONUS, most notably, due to differing emissions profiles and the prevalence of wildfire smoke in the West. Therefore, in addition to geographic coordinates being out of distribution, the target value follows different dynamics too. While understanding these caveats, we can still compare models with no geolocation features against the ones enhanced with fused GeoCLIP embeddings.

For the Experiment 1 group (train on West, test on East), the model with GeoCLIP embeddings outperforms the model without lat/lon, achieving a higher average R^2 of 0.47 (vs. 0.38) and a lower average RMSE of 6.90 (vs. 7.43 $\mu g/m^3$). Similarly, in the Experiment 2 group (train on East, test on West), the GeoCLIP model again leads with an R^2 of 0.61 (vs. 0.52) and RMSE of 3.21 (vs. 3.56 $\mu g/m^3$). In both cases, the GeoCLIP-encoding-augmented models exhibit better generalization to the geographically disjoint test regions. These results further demonstrate the effectiveness of using pretrained GeoCLIP embeddings in enhancing out-of-region performance, even under significant spatial disjointedness and differing aerosol dynamics between the western and eastern U.S. The embeddings likely encode semantic geographic context that aids the model in adapting to previously unseen regions. Despite the distribution shift and spatial disjointedness, GeoCLIP embeddings continue to provide semantically meaningful geographic context that boosts predictive performance across CONUS-wide OoR scenarios.

3.3. Ablation

There are currently not many pretrained location encoders available, as this is a burgeoning area of research. We considered the ones available, including SatCLIP [10], GeoCLIP [11], and CSP [9] . For our experiments we considered using location encoders of GeoCLIP and SatCLIP, given that in [10] evaluations, CSP showed considerably lower comparative performance. However, it must be noted that CSP was one of the pioneer papers in this field.

GeoCLIP uses Fourier features for its positional encoder, has an embedding dimension of 512, and uses Flickr images to distill information into location embeddings, which consist of natural imagery taken by ordinary users. SatCLIP, on the other hand, use spherical harmonics, has an embedding space of dimension 256, and more importantly, uses Sentinel-2 satellite imagery to train the model, with a focus on global sampling of the images and locations.

There are two general strategies for fusing pre-learned location embeddings with other predictive features. The downstream model $f(\cdot)$ then learns to combine these embeddings with task-specific features for improved predictions, while benefiting from the spatial knowledge that was already encapsulated in e_s. In our deep learning pipeline

for estimating daily $PM_{2.5}$ concentrations, the location embedding e_s is fused with the temporal feature representation e_{ts} output by the Bi-LSTM. A typical fusion strategy involves either concatenation:

$$y = f\left(e_{\text{ts}} \oplus e_s\right)$$

or elementwise (Hadamard) product:

$$y = f\left(e_{\text{ts}} \odot e_s\right)$$

where $f(\cdot)$ denotes the final prediction head (e.g., a fully connected layer). Figure 5 shows the results of experiments with both GeoCLIP and SatCLIP using both fusion strategies.

Figure 5. Ablation results on the random test set. Hadamard fusion strategy performs the best for location encoders, and GeoCLIP notably outperforms SatCLIP for our task.

While GeoCLIP shows superior performance compared to SatCLIP (Figure 5), it is worth remembering that these are not production-level models yet, and rather, research products. For instance, the embedding dimension of 512 for GeoCLIP versus the 256 dimension for SatCLIP may impact their performance in our task, rather than the underlying positional encoders in each or training strategies. As more research is dedicated to location encoders in the near future, we expect more integration and evaluation in real-world tasks. As it relates to our task, however, it is worth discussing the underlying data of each location encoder. GeoCLIP *may* outperform SatCLIP for $PM_{2.5}$ estimation tasks because its location embeddings are pretrained on Flickr imagery, which often captures ground-level human-centric scenes, including urban infrastructure, roads, and pollution-emitting facilities. These features are visually prominent and related with $PM_{2.5}$ sources, allowing the model to encode relevant local context. In contrast, SatCLIP relies on Sentinel-2 imagery, which captures land cover from a top-down satellite perspective and *may* miss finer-grained indicators of

anthropogenic pollution sources (e.g., industrial stacks, traffic congestion), leading to less informative embeddings for this application.

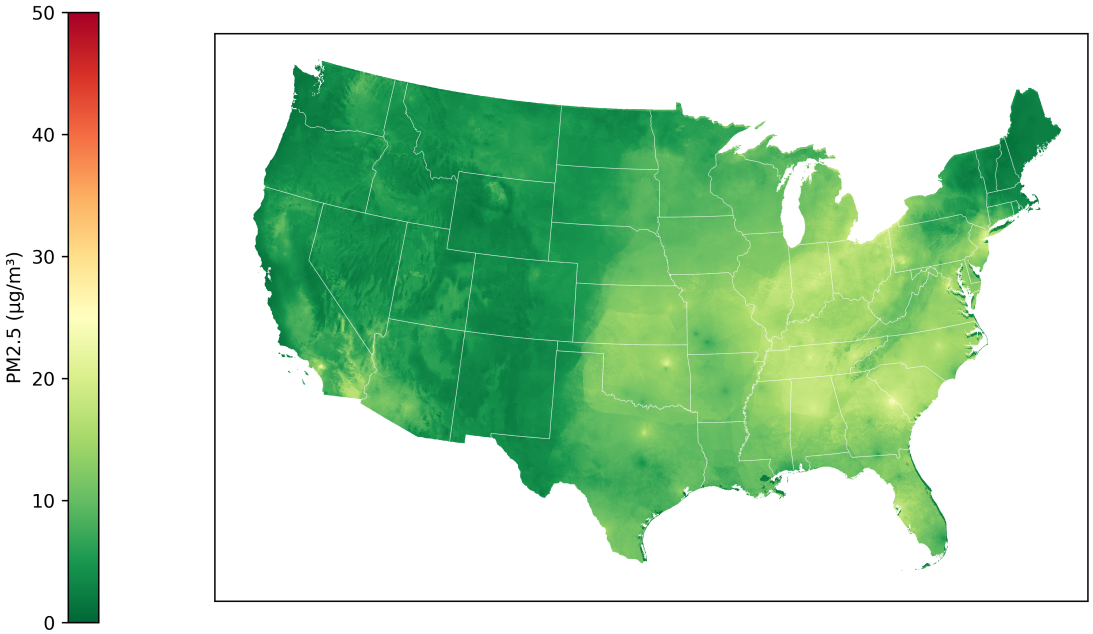
3.4. Qualitative Analysis of Spatial Patterns

While the summary performance metrics provided in the previous sections shed light on the overall estimation performance of the models, we can visualize the model output over space. However, it is worth remembering that there is no 'ground truth' available for the entire space, i.e., there is no satellite can observe the surface-level $PM_{2.5}$ concentrations at all locations against which the model output can be compared. Figure 6 provides a comparison of the maps generated using two different models, one without any geolocation feature (a), and (b) with pretrained GeoCLIP features, which performed best in the OOR and WR evaluation scenarios.

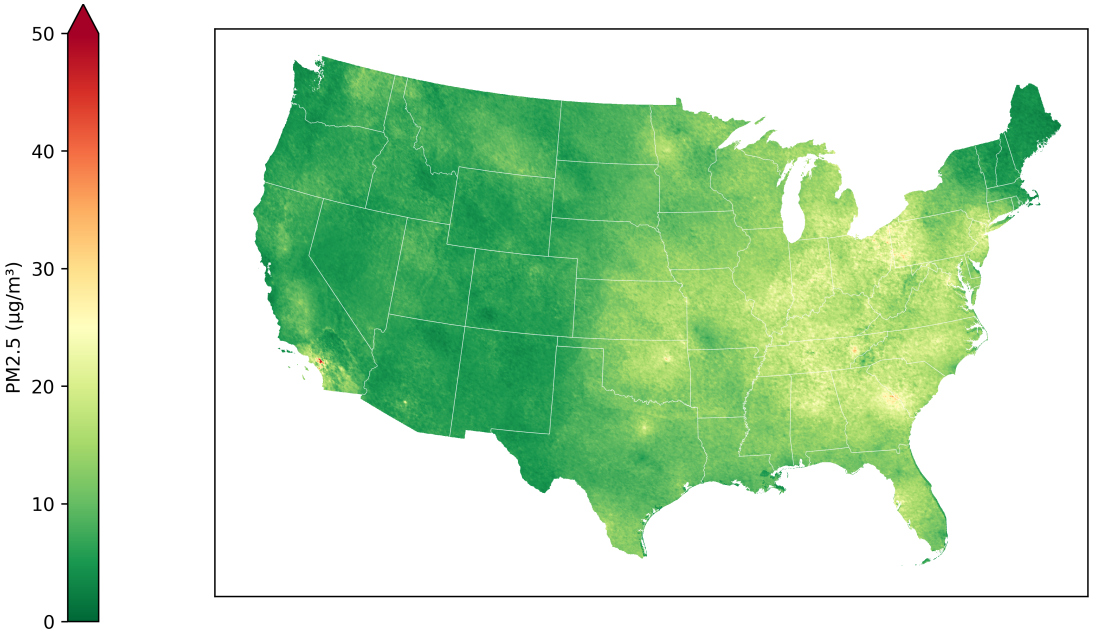
Visual inspection of predicted $PM_{2.5}$ concentration maps (Figure 6) reveals differences between models with and without geolocation features, and reminds of us of the spatial distribution of (a) pretraining imagery used for GeoCLIP pretraining, and (b) AQS stations used in training the downstream fused Bi-LSTM. However, these manifest in the outcome very differently.

The GeoCLIP-based model exhibits smoother and more spatially coherent patterns in several areas such the Great Plains States, where the model without geolocation tends to produce blocky artifacts with sharp an unnatural transitions, likely due to its inability to generalize across regions with sparse AQS measurement used in downstream Bi-LSTM training. This suggests that GeoCLIP's pretrained location embeddings may help encode broad spatial context, enabling better geographic generalization. Interestingly, and of high importance to the air pollution estimation task (which is primarily intended to support health impact quantification in urban areas), in several urban centers—including Los Angeles, CA; Augusta, GA; Tulsa, OK; Pittsburgh, PA; and Cleveland, OH, all cities with known $PM_{2.5}$ pollution. The GeoCLIP-fused model captures elevated $PM_{2.5}$ levels at these known pollution hotspots, saturating the map color scale, rendering regions visibly red. The model without geolocation features seem to have smoother and lower value outputs in these places, even in Southern California. The fact that the model fused with GeoCLIP pre-learned embeddings is able to estimate higher values in urban areas counters a potential concern that spatial smoothing of static embeddings and reliance on static features would suppress (temporally dynamic) extremes, and highlights that the pretrained encoder can enhance urban prediction. This is in line with the quantitative results presented in the previous sections, where summary metrics pointed to the higher performance of GeoCLIP-fused models.

On the other hand, the GeoCLIP output map seems to contain speckling or gaussian noise-like patterns in some rural or topographically complex areas. We believe this might be due to a combination of factors. First, the use of high-frequency Fourier features in the positional encoder can introduce localized instability, particularly when the downstream decoder $g(\phi(\varphi, \lambda))$ is applied to areas lacking strong pretraining samples. This is analogous to overfitting with high-degree basis functions, where unregularized regions may yield high-variance outputs. Put differently, the high-frequency positional encodings give the model the *flexibility* to encode fine-scale spatial variation, but in regions lacking training signal, this flexibility can lead to high-variance or noisy predictions. Second, the underlying Flickr imagery used in GeoCLIP's training is spatially uneven: denser in urban or tourist-heavy locations and sparser elsewhere. This uneven



(a) Without geolocation features



(b) With pretrained location encoder (GeoCLIP)

Figure 6. Qualitative comparison of predicted $PM_{2.5}$ concentrations on July 5, 2021 across the contiguous U.S. Top (a): estimation using a model without geolocation features. Bottom (b): estimation using fused pretrained GeoCLIP location embeddings. Each map is generated at a spatial resolution of 1 km.

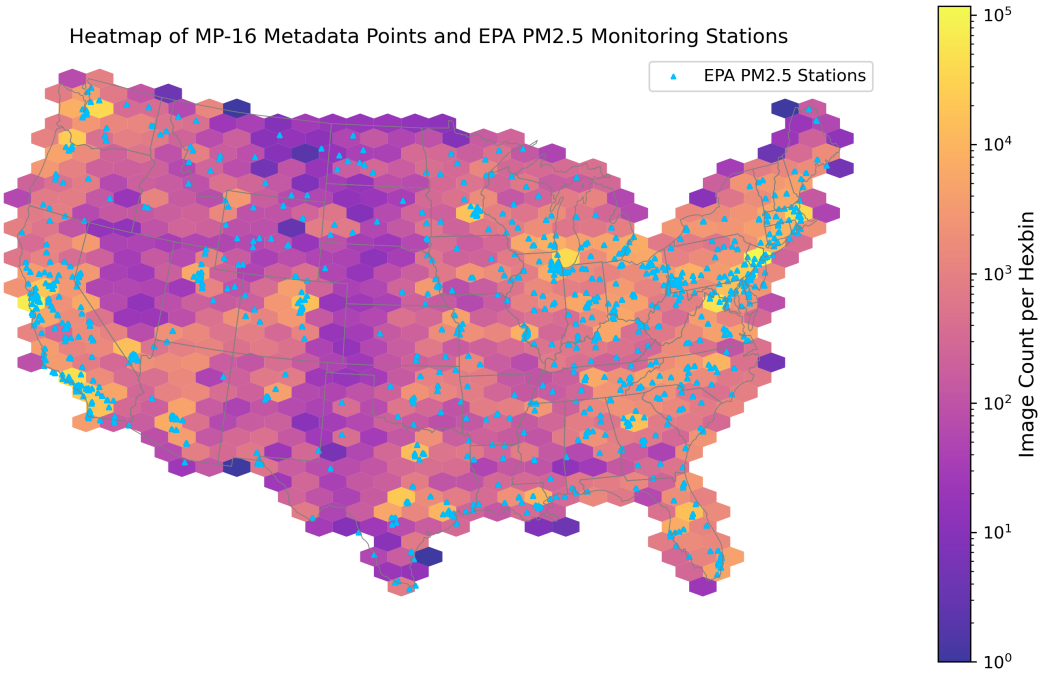


Figure 7. Spatial distribution of Flickr image density used in the MP-16 dataset (used for pretraining GeoCLIP [11]) overlaid with U.S. EPA $PM_{2.5}$ AQS monitoring stations. Flickr density is shown using a hexbin log-scale colormap; EPA monitoring stations are marked with cyan triangles. Both have higher densities in higher population areas.

distribution may bias the learned representations and reduce embedding quality in under-sampled regions. Furthermore, fine-scale terrain-driven variations (e.g., along the Rockies) are more blurred in the GeoCLIP output. This may suggest that the model is not fully leveraging elevation or microclimate variation in those regions when fused with a static embedding.

Figure 7 illustrates the spatial distribution of the Flickr images used in the MP-16 dataset—which are used in pretraining GeoCLIP embeddings—alongside the U.S. EPA $PM_{2.5}$ AQS monitoring network. It can generally be seen that the more populated an area is, the more AQS stations and the higher the density of Flickr pretraining images are (e.g., the Northeast, Southern California). The reason for higher density of Flickr images in populated areas is rather obvious: these are crowdsources, human-captured imagery. As for the spatial distribution of AQS stations, in the United States, $PM_{2.5}$ monitors operated by the U.S. Environmental Protection Agency (EPA) are used to assess compliance with the National Ambient Air Quality Standards (NAAQS)—federally mandated thresholds for ambient air pollutants intended to protect public health and the environment. Because most areas that are either currently designated as nonattainment or at risk of becoming nonattainment are urban and suburban, the majority of monitoring stations are concentrated in these regions. However, since the NAAQS also apply to Class I areas such as National Parks and Wilderness Areas, some monitors are sited in remote or protected landscapes to ensure compliance in those settings. Overall, sparsely monitored regions generally correspond to sparsely populated areas, including the Great Plains from North Dakota through West Texas, and the interior Southwest from Nevada through New Mexico.

This general alignment in spatial distributions partly explains the improvement in

the overall estimation performance metrics as well as improved predictions of higher $PM_{2.5}$ concentration values in urban region. However, other regions such as the Mountain West exhibit Flickr image density sparsity (for pretraining) despite the presence of AQS monitors (for downstream task training). This spatial mismatch explains the limitation observed above, and highlights a limitation of pretrained location encoders: uneven coverage in the upstream pretraining dataset may result in lower-quality or biased embeddings in rural or topographically complex areas, potentially contributing to the spatial noise observed in those regions.

These qualitative analyses show the advantage and risks of location encoders pretrained on sparse data: they can enhance specificity and realism in familiar, well-sampled regions, but also amplify artifacts where spatial signal is weak or noisy, particularly when higher-frequency basis functions (such as Fourier features) are used in position encoding. Future research can perhaps focus on adaptive adjustment of positional encoder frequency degrees, particularly in lower-sampled regions.

3.5. Evaluation Under High-Concentration Conditions: The 2021 Dixie Fire

The 2021 Dixie Fire was one of the largest and most destructive wildfires in California history, igniting on July 13, 2021 in the northern Sierra Nevada, near the Cresta Dam in Butte County. It scorched approximately 963,000 acres across five counties (Butte, Plumas, Tehama, Lassen, and Shasta). The fire destroyed nearly 1,300 structures and generated sustained high levels of $PM_{2.5}$ across large areas of the western United States. Figure 8 compares the daily $PM_{2.5}$ estimations of the baseline model (without geolocation features) and the GeoCLIP-enhanced model over four dates spanning peak fire activity. In each case, columns (a) and (b) depict the baseline model output, while columns (c) and (d) show the GeoCLIP-augmented model output at CONUS and regional scales. Notably, the GeoCLIP-enhanced model produces larger, contiguous plumes with higher $PM_{2.5}$ concentrations over Northern California, better aligning with known fire-affected areas. In contrast, the baseline model exhibits more spatial fragmentation and muted intensity, noticeably underestimating the scale and magnitude of pollution peaks. We nevertheless acknowledge the persistence of potential artifacts, for instance, along NW to SE direction for the GeoCLIP-fused model, or in the case of the fourth row of Figure 8, the relatively high values (light yellow) in disconnected patches surrounding the fire centers. In comparison, baseline model predictions look smoother across space, however, at the potential cost of underestimation at some high concentration areas.

Figure 9 further visualizes the spatial residuals for the same dates alongside NOAA HMS smoke density polygons. The GeoCLIP-enhanced model reduces systematic negative bias in smoke-affected regions, as seen in the transition from large blue residuals (underprediction) in the baseline model to more neutral residuals. This indicates an improved capacity to capture elevated pollution levels during dynamic wildfire events.

These observed patterns are consistent with the previous summary metrics and single-day maps: the GeoCLIP-fused model not only improves generalization across space, but also enhances larger value estimates under extreme air quality conditions. By encoding location semantics via pretrained embeddings, the model is better positioned to estimate higher concentrations in regions lacking dense observations but

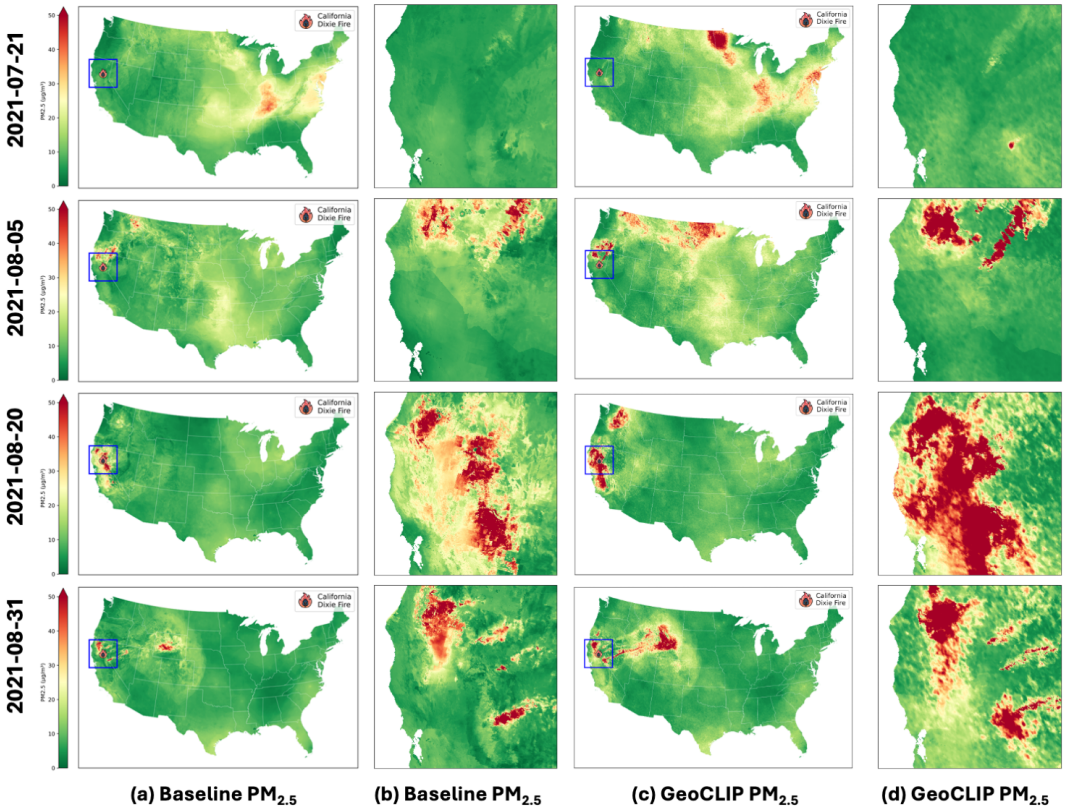


Figure 8. Estimated $PM_{2.5}$ during the 2021 Dixie Fire (Northern California) using baseline (with no geographic features) and GeoCLIP-enhanced models. Each row shows a different date during peak wildfire activity. Left columns (a, b): baseline model. Right columns (c, d): GeoCLIP-fused model. The GeoCLIP-enhanced model predictions show stronger plume intensity and spatial coherence, particularly in fire-affected regions.

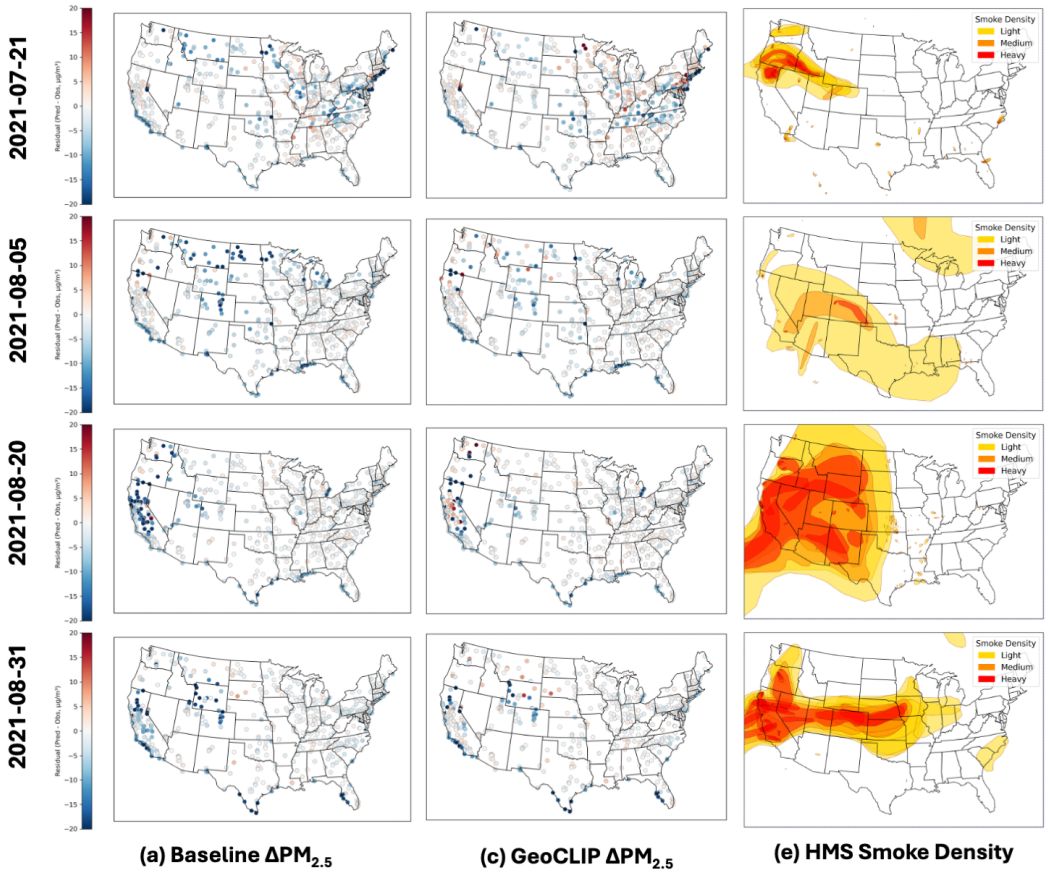


Figure 9. Residuals of $PM_{2.5}$ estimations (model estimate minus observed) during the 2021 Dixie Fire, with NOAA HMS smoke plumes overlaid. Baseline model which does not use geographic features (left) shows systematic underprediction (blue) in fire regions. GeoCLIP-enhanced model (middle) reduces negative residuals, and aligns better with the smoke plumes. Rightmost column shows HMS smoke density for visual reference.

exhibiting complex pollution dynamics—such as those during wildfire plumes.

Despite improved performance, the GeoCLIP-enhanced model still underpredicts $PM_{2.5}$ in smoke-affected areas. This residual bias may stem from limited downstream training data in extreme pollution conditions, the static nature of GeoCLIP embeddings (which cannot reflect evolving fire behavior or atmospheric transport), and uneven upstream pretraining coverage. Fine-scale variability due to complex terrain or plume lofting (e.g., over the Sierra Nevada) may also be poorly captured by models relying on coarser inputs. These results suggest that while pretrained location embeddings improve spatiotemporal estimates, further gains may require integrating further contextual features into location encoders, as well as more strategies for enhancing extreme value estimations in downstream training.

4. Discussion

4.1. Estimation Performance Comparison

We built on a previously validated Bi-LSTM with Attention model that achieves state-of-the-art performance in $PM_{2.5}$ estimation, including during high-concentration events [25]. Compared to the Di et al. [23] dataset from 2005 to 2016, our model improves RMSE from 2.73 to 2.63 (-3.7%) for all days, and 19.01 to 15.44 (-18.8%) for high concentration days (>35 $\mu g/m^3$). Compared to Wei et al. [28] dataset which covers 2017 to 2021, for the same period, our base Bi-LSTM model improves the RMSE from 4.70 to 2.73 (-41.9%) for all days, and 26.45 to 20.78 (-21.4%) for high concentration days (>35 $\mu g/m^3$). Our Bi-LSTM with Attention explains 10% more variance than our own Random Forest strong baseline, and 4% more than a regular LSTM; RMSE is 3.59 (vs RF's 4.00, -10.3%, and LSTM's 3.81, -5.8%). More details and comparisons of the underlying air pollution estimation model can be found in [25].

Using this strong model enabled us to focus specifically on the added value of geolocation features, without conflating gains from broader architectural or modeling differences. By building on an underlying model that already performs competitively against existing air pollution datasets, we strengthen the validity of our findings regarding the role of geolocation in enhancing generalizability and spatial transferability.

4.2. On the Impact of Coordinate Transformation Methods

In our experiments, for naïve incorporation of geolocation features (Approach 2), we applied sinusoidal wrapping to encode latitude and longitude. This approach preserves angular continuity while mapping coordinates into a bounded feature space. A natural question is whether alternative wrapping methods—such as those cataloged in [8] are necessary. Our evaluation shows that the impact of coordinate wrapping on generalization and performance is largely invariant across such encodings (e.g., with raw coordinates vs. sinusoidal wrapping). The contiguous U.S. spans a continuous and bounded range of approximately 24°–49° N and 67°–125° W, without spatial discontinuities like poles or the International Date Line that would otherwise necessitate more complex encodings. Within this extent, all reasonable wrapping strategies—including sinusoidal transforms and Fourier projections—preserve relative spatial locality in similar ways. More importantly, our results show that the dominant factor influencing geographic generalizability is not the coordinate transformation itself, but whether coordinates are naïvely passed to the model or replaced with pretrained higher-dimension embed-

dings such as those from GeoCLIP. Put differently, while the choice of encoding may affect inductive bias slightly, its influence is eclipsed by the broader modeling approach to incorporating geolocation.

Another question might be whether more expressive strategies like Fourier feature expansion [57]—would improve model performance, and more importantly, geographic generalizability.

The higher-frequency components of Fourier Expansion enable neural networks to learn high-frequency variations (i.e., higher spatial resolutions) in the input space by making the mapping more expressive, capturing higher resolution variations.

From a domain perspective, our target variable (daily $PM_{2.5}$ concentrations) and its geophysical covariates (e.g., meteorology, AOD, land cover) exhibit smooth spatial variation across the contiguous U.S. Additionally, since our model also incorporates temporal context and powerful sequence modeling (Bi-LSTM with Attention), the marginal impact of switching to a more expressive wrapping method on estimation performance is negligible. However, a more expressive wrapping method is sure to increase model capacity, which in turn, will inevitably hurt geographic generalizability, for similar reasons that we explain in the next paragraph.

4.3. Generalized Location Representations vs. Fine-tuning the Location Encoder

Another related question is whether fine-tuning location encoders such as GeoCLIP on the downstream $PM_{2.5}$ estimation task would lead to performance gains. However, we intentionally avoided fine-tuning the location encoder to preserve and examine the generalization benefits that pretrained encoders are designed to provide. Fine-tuning poses two major risks in this context, defeating the purpose of using pretrained location encoders. First, the positional encoders that feed into location encoders (e.g., Fourier features with k frequency bands) yield a much higher-dimensional representation than raw latitude and longitude (e.g., 16–64 features versus 2 or 4), substantially increasing model capacity and thus the risk of overfitting—especially when training data is sparse or regionally biased. This effect is exacerbated in our case where the downstream model must learn from limited and sparse $PM_{2.5}$ ground truth. For this same reason, a more expressive location wrapping method (such as Fourier features) will undoubtedly be more hurtful to geographic generalizability in the direct incorporation of location features (e.g., Approach 2) as well.

Second, if the location encoder $g(\phi(\varphi, \lambda))$ is made trainable in the downstream task, it may begin to encode spurious location-specific patterns or implicitly compensate for missing covariates (e.g., unobserved emission sources or meteorological anomalies), which can entangle the learned representations with nuisance factors that do not generalize beyond the training spatial extent. This would undermine the very goal of location encoders: to produce generalized, reusable, and disentangled geographic representations distilled from large, unlabeled datasets during pretraining. Freezing the encoder ensures that the spatial attributes extracted from pretraining (e.g., Flickr imagery) remain invariant and generalize to new tasks. As our results confirm, using fixed embeddings from GeoCLIP improves both within-region and out-of-region performance—highlighting that the benefits of pretrained location encoders can stem from their consistency, not their adaptability to downstream noise or task peculiarities.

5. Conclusion

In this paper, we quantified the impacts of incorporating geolocation information in deep learning for temporally-dynamic estimation, with emphasis on geographic generalizability. We conducted a series of experiments by modifying a model for estimating daily average concentrations of $PM_{2.5}$ in the continental United States. We measured and compared within-region (WR) and Out-of-region (OoR) performance by incorporating geographic coordinates as features, as well as incorporating location embeddings from a more advanced GeoCLIP location encoder [11]. Our WR evaluations intended to examine whether incorporating geolocation information into deep learning enhances its interpolative ability (where test locations are roughly within the bounds of training locations). Our results indicated that adding geographic coordinates as features indeed appear to add interpolative value to the estimation task, while embeddings from the GeoCLIP location encoder even further enhanced WR evaluation performance.

We also conducted experiments to evaluate OoR performance, by partitioning the geographic area into non-overlapping, disjoint areas that separate training and testing data. Our results highlight that naïve inclusion of geographic coordinates as features can hinder model performance in OoR scenarios; however, well-designed location encoders such as GeoCLIP provide improvements in geographic generalizability. These findings show how geolocation features influence model behavior. In WR scenarios, geographic coordinates can serve as useful disambiguators—helping the model refine predictions when observations are spatially proximate. However, in OoR scenarios, these same features encourage overfitting by allowing the model to learn location-target associations rather than generalizable mappings from observations to target. This contrast highlights the value of incorporating location in ways that prioritize observational relevance rather than region-specific memorization.

In addition to quantitative results, we provided an in-depth discussion of the methods of incorporating geolocation into deep learning, along with analyses of variations and ablations, and the expected impacts on geographic generalizability. Improved geographic generalizability in $PM_{2.5}$ estimation has direct importance for public health and environmental policy. Models that can accurately estimate pollution levels across both data-rich and data-sparse regions support more equitable exposure assessments, especially in under-monitored or rural communities. This enhances the ability of policymakers to implement data-driven air quality regulations, evaluate compliance with federal standards, and better allocate resources for monitoring and mitigation efforts.

While the original intention of our work was to quantify the impacts of incorporating geolocation information into deep learning for the estimation of a target with day-to-day changes with multi-variate observations, our results also highlight the value of location encoders (despite and in addition to the multi-variate domain observations), and the value of an emerging research body to distill information about the Earth in latent embeddings for seamless incorporation into deep learning. This area of research is relatively young, and so far, has been focused on fusing one mode of data with location representations. To distill even more information into these models, future research can move towards multi-modal and multi-sensor enhancements that better capture information about a place into these latent embeddings. As evidenced by our dynamic application that requires daily observation of aerosol content in the atmosphere, adding a temporal aspect to location encoders can also be invaluable in applications on dynamic phenomena.

Further, our qualitative analysis revealed the impact of spatially-uneven pretraining samples and spatially-sparse downstream supervision values, resulting in noise artifacts

or smoothed-out estimates in certain regions. The uneven distribution of pretraining samples combined with high-degree basis functions used in positional encoders of location encoders appears to result in speckle-like noise. Therefore, spatial regularization of positional encoders with adaptive scales promises to be a productive area for future research.

Data Availability Statement

All data uses are openly available at [58] and all software code are available at [59].

Funding

The National Science Foundation grant number 2026962 and NIH/NIEHS grant R21ES032973 have supported this work. The content is solely the responsibility of the authors and does not necessarily represent the official views of the University of Colorado, NSF, NIH, or NIEHS.

Author Contributions

Morteza Karimzadeh: Conceptualization, Methodology, Formal analysis, Writing – original draft, Supervision.
Zhongying Wang: Data curation, Software, Validation, Methodology, Writing – review & editing.
James L. Crooks: Resources, Supervision, Methodology, Writing – review & editing.

Disclosure statement

Authors have no conflict of interest to report.

References

[1] Tobler WR. A computer movie simulating urban growth in the detroit region. Economic geography. 1970;46(sup1):234–240.
[2] Anselin L. Spatial econometrics: methods and models. Vol. 4. Springer Science & Business Media; 2013.
[3] Matheron G. Principles of geostatistics. Economic geology. 1963;58(8):1246–1266.
[4] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: Advances in Neural Information Processing Systems; Vol. 30; Curran Associates, Inc.; 2017. p. 5998–6008.
[5] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems. 2012;25.
[6] Hochreiter S. Long short-term memory. Neural Computation MIT-Press. 1997;.
[7] LeCun Y, Bengio Y, Hinton G. Deep learning. nature. 2015;521(7553):436–444.
[8] Mai G, Janowicz K, Hu Y, et al. A review of location encoding for geoai: methods and applications. International Journal of Geographical Information Science. 2022;36(4):639–673.

[9] Mai G, Lao N, He Y, et al. Csp: Self-supervised contrastive spatial pre-training for geospatial-visual representations. In: International Conference on Machine Learning; PMLR; 2023. p. 23498–23515.
[10] Klemmer K, Rolf E, Robinson C, et al. Satclip: Global, general-purpose location embeddings with satellite imagery. arXiv preprint arXiv:231117179. 2023;.
[11] Vivanco Cepeda V, Nayak GK, Shah M. Geoclip: Clip-inspired alignment between locations and images for effective worldwide geo-localization. Advances in Neural Information Processing Systems. 2023;36:8690–8701.
[12] Lucas B, Vahedi B, Karimzadeh M. A spatiotemporal machine learning approach to forecasting covid-19 incidence at the county level in the usa. International Journal of Data Science and Analytics. 2023;15(3):247–266.
[13] Radford A, Kim JW, Hallacy C, et al. Learning transferable visual models from natural language supervision. In: International conference on machine learning; PMLR; 2021. p. 8748–8763.
[14] Rolf E, Proctor J, Carleton T, et al. A generalizable and accessible approach to machine learning with global satellite imagery. Nature communications. 2021;12(1):4392.
[15] Wang J, Hopkins L, Hallman T, et al. Cross-validation for geospatial data: Estimating generalization performance in geostatistical problems. Transactions on Machine Learning Research. 2023;.
[16] Wang L, Adiga A, Chen J, et al. Causalgnn: Causal-based graph neural networks for spatio-temporal epidemic forecasting. In: Proceedings of the AAAI conference on artificial intelligence; Vol. 36; 2022. p. 12191–12199.
[17] Zhou S, Wang W, Zhu L, et al. Deep-learning architecture for pm2. 5 concentration prediction: A review. Environmental Science and Ecotechnology. 2024;:100400.
[18] Qin D, Yu J, Zou G, et al. A novel combined prediction scheme based on cnn and lstm for urban pm2.5 concentration. IEEE Access. 2019;7:20050–20059.
[19] Zhao G, He H, Huang Y, et al. Near-surface pm2. 5 prediction combining the complex network characterization and graph convolution neural network. Neural Computing and Applications. 2021;33(24):17081–17101.
[20] Kim DY, Jin DY, Suk HI. Spatiotemporal graph neural networks for predicting mid-to-long-term pm2. 5 concentrations. Journal of Cleaner Production. 2023;425:138880.
[21] Yu M, Masrur A, Blaszczak-Boxe C. Predicting hourly pm2. 5 concentrations in wildfire-prone areas using a spatiotemporal transformer model. Science of The Total Environment. 2023;860:160446.
[22] Pathak RS, Pathak V, Rai A. A novel attention-based deep learning model for accurate pm2. 5 concentration prediction and health impact assessment. Journal of Atmospheric and Solar-Terrestrial Physics. 2025;:106583.
[23] Di Q, Amini H, Shi L, et al. An ensemble-based model of pm2. 5 concentration across the contiguous united states with high spatiotemporal resolution. Environment international. 2019;130:104909.
[24] Ryan PH, LeMasters GK. A review of land-use regression models for characterizing intraurban air pollution exposure. Inhalation toxicology. 2007;19(sup1):127–133.
[25] Wang Z, Crooks JL, Regan EA, et al. High-resolution estimation of daily pm2. 5 levels in the contiguous us using bi-lstm with attention. Remote Sensing. 2025;17(1):126.
[26] Rolph GD, Draxler RR, Stein AF, et al. Description and verification of the noaa smoke forecasting system: the 2007 fire season. Weather and Forecasting. 2009;24(2):361–378.
[27] McNamara D, Stephens G, Ruminski M, et al. The hazard mapping system (hms)–noaa multi-sensor fire and smoke detection program using environmental satellites. In: 13th conference on satellite meteorology and oceanography; Vol. 22; 2004.
[28] Wei J, Li Z, Lyapustin A, et al. First close insight into global daily gapless 1 km pm2. 5 pollution, variability, and health impact. Nature communications. 2023;14(1):8349.
[29] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE. 1998;86(11):2278–2324.
[30] Zhu XX, Tuia D, Mou L, et al. Deep learning in remote sensing: A comprehensive review

and list of resources. IEEE Geoscience and Remote Sensing Magazine. 2017;5(4):8–36.
[31] de Lima RP, Karimzadeh M. Model ensemble with dropout for uncertainty estimation in sea ice segmentation using sentinel-1 sar. IEEE Transactions on Geoscience and Remote Sensing. 2023;61:1–15.
[32] Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. Nature. 1986;323(6088):533–536.
[33] Zhu XX, Tuia D, Mou L, et al. Deep learning in remote sensing: A comprehensive review and list of resources. IEEE Geoscience and Remote Sensing Magazine. 2017;5(4):8–36.
[34] Shi X, Chen Z, Wang H, et al. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In: Advances in Neural Information Processing Systems; Vol. 28; Curran Associates, Inc.; 2015. p. 802–810.
[35] Le VD, Bui TC, Cha SK. Spatiotemporal deep learning model for citywide air pollution interpolation and prediction. In: 2020 IEEE international conference on big data and smart computing (BigComp); IEEE; 2020. p. 55–62.
[36] Kang J, Jiang W, Xu W, et al. Geospatial transformer for satellite imagery. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW); IEEE; 2020. p. 300–301.
[37] Song X, Hua Z, and JL. Gmts: Gnn-based multi-scale transformer siamese network for remote sensing building change detection. International Journal of Digital Earth. 2023; 16(1):1685–1706. Available from: https://doi.org/10.1080/17538947.2023.2210311.
[38] Sutskever I. Sequence to sequence learning with neural networks. arXiv preprint arXiv:14093215. 2014;.
[39] Stokholm A, Buus-Hinkler J, Wulf T, et al. The autoice competition: Automatically mapping sea ice in the arctic. In: EGU General Assembly Conference Abstracts; 2023. p. EGU–13038.
[40] Chen X, Patel M, Pena Cantu FJ, et al. Mmseaice: a collection of techniques for improving sea ice mapping with a multi-task model. The Cryosphere. 2024;18(4):1621–1632.
[41] Reid CE, Considine EM, Maestas MM, et al. Daily pm2. 5 concentration estimates by county, zip code, and census tract in 11 western states 2008–2018. Scientific data. 2021; 8(1):112.
[42] Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional lstm and other neural network architectures. Neural networks. 2005;18(5-6):602–610.
[43] Schuster M, Paliwal KK. Bidirectional recurrent neural networks. IEEE transactions on Signal Processing. 1997;45(11):2673–2681.
[44] Luong MT. Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:150804025. 2015;.
[45] Windsor H, Toumi R. Scaling and persistence of uk pollution. Atmospheric environment. 2001;35(27):4545–4556.
[46] Lyapustin A, Wang Y, Korkin S, et al. Modis collection 6 maiac algorithm. Atmospheric Measurement Techniques. 2018;11(10):5741–5765.
[47] Thornton PE, Thornton MM, Mayer BW, et al. Daymet: Daily surface weather data on a 1-km grid for north america, version 2. Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States); 2014.
[48] Abatzoglou JT. Development of gridded surface meteorological data for ecological applications and modelling. International journal of climatology. 2013;33(1):121–131.
[49] Danielson JJ, Gesch DB. Global multi-resolution terrain elevation data 2010 (gmted2010). US Geological Survey; 2011.
[50] Didan K, Munoz AB, Solano R, et al. Modis vegetation index user's guide (mod13 series). University of Arizona: Vegetation Index and Phenology Lab. 2015;35:2–33.
[51] Wang Z, de Lima RP, Crooks JL, et al. Increasing the spatial coverage of atmospheric aerosol depth measurements using random forest and mean filters. In: IGARSS 2023 - 2023 IEEE International Geoscience and Remote Sensing Symposium; 2023. p. 3928–3931.
[52] Che Z, Purushotham S, Cho K, et al. Recurrent neural networks for multivariate time series with missing values. Scientific reports. 2018;8(1):6085.

[53] Ba JL. Layer normalization. arXiv preprint arXiv:160706450. 2016;.
[54] Kingma DP. Adam: A method for stochastic optimization. arXiv preprint arXiv:14126980. 2014;.
[55] Sutskever I, Martens J, Dahl G, et al. On the importance of initialization and momentum in deep learning. In: International conference on machine learning; PMLR; 2013. p. 1139–1147.
[56] Huber PJ, Ronchetti EM. Robust statistics. John Wiley & Sons; 2011.
[57] Tancik M, Srinivasan P, Mildenhall B, et al. Fourier features let networks learn high frequency functions in low dimensional domains. Advances in neural information processing systems. 2020;33:7537–7547.
[58] Karimzadeh M, Wang Z, Crooks JL. Contiguous u.s. daily pm2.5 measurements (2021) – benchmark dataset for location-encoder evaluation ; 2025. Dataset.
[59] Karimzadeh M, Wang Z, Crooks JL. pm25-geolocation: Codebase for geolocation-aware pm2.5 estimation [https://github.com/geohai/pm25-geolocation]; 2025. Version 1.0, Accessed: 2025-08-04.

Appendix A. Evaluation Metrics Tables

Table A1. Within-Region evaluation metrics for five random splits, each with a different training and testing subset. Mean ± standard deviation are reported. Variation in metrics over the five splits reflects differences in subset difficulty and target distributions.

Model variant	**Test** R^2	**Test RMSE** (μg m^{-3})	**Test MBE** (μg m^{-3})
Without Lat/Lon	0.73 ± 0.035	3.80 ± 0.24	-0.06 ± 0.72
With Lat/Lon	0.74 ± 0.016	3.79 ± 0.11	0.07 ± 0.79
With Sinusoidal (Lat/Lon)	0.75 ± 0.024	3.69 ± 0.17	0.11 ± 0.70
GeoCLIP Location Embeddings (Hadamard)	0.79 ± 0.010	3.41 ± 0.08	-0.03 ± 0.32
GeoCLIP Location Embeddings (Concat)	0.73 ± 0.036	3.81 ± 0.25	0.26 ± 0.93
SatCLIP Location Embeddings (Hadamard)	0.68 ± 0.077	4.15 ± 0.45	0.39 ± 1.31
SatCLIP Location Embeddings (Concat)	0.72 ± 0.024	3.89 ± 0.16	-0.17 ± 0.73

Table A2. Within-Region evaluation metrics for five different spatial splits (with entire time series of test stations held out), each with a distinct training and testing subset. Mean ± standard deviation are reported. Variation across splits reflects differences in spatial coverage and target distributions.

Model variant	**Test** R^2	**Test RMSE** (μg m^{-3})	**Test MBE** (μg m^{-3})
Without Lat/Lon	0.62 ± 0.13	4.46 ± 0.81	0.05 ± 1.63
With Lat/Lon	0.64 ± 0.067	4.37 ± 0.71	-0.16 ± 1.04
With Sinusoidal (Lat/Lon)	0.65 ± 0.074	4.27 ± 0.64	-0.36 ± 0.87
GeoCLIP Location Embeddings	0.67 ± 0.049	4.21 ± 0.65	0.28 ± 0.47

Table A3. Out-of-Region (Checkerboard, δ=8°): metrics on the two disjoint test partitions 1 and 2. The partitions contain different proportions of test stations—48.59% in partition 1 and 51.41% in partition 2—reflecting the inherently uneven geographic distribution of AQS monitoring sites across the contiguous United States.

Model variant	**Test R^2 (1|2)**	**Test RMSE (μg m^{-3}) (1|2)**	**Test MBE (μg m^{-3}) (1|2)**
Without Lat/Lon	0.59 | 0.53	4.02 | 5.61	-1.00 | 0.27
With Lat/Lon	0.58 | 0.47	4.06 | 5.98	-0.51 | -1.42
With Sinusoidal (Lat/Lon)	0.58 | 0.47	4.04 | 5.99	0.42 | -0.65
GeoCLIP Location Embeddings	0.60 | 0.54	3.97 | 5.53	0.19 | 0.19

Table A4. Out-of-Region (Checkerboard, δ=16°): metrics on the two disjoint test partitions 1 and 2. The partitions contain different proportions of test stations—63.51% in partition 1 and 36.49% in partition 2—reflecting the inherently uneven geographic distribution of AQS monitoring sites across the contiguous United States.

Model variant	**Test R^2 (1|2)**	**Test RMSE (μg m^{-3}) (1|2)**	**Test MBE (μg m^{-3}) (1|2)**
Without Lat/Lon	0.46 | 0.64	5.62 | 4.01	-1.68 | -0.15
With Lat/Lon	0.46 | 0.59	5.62 | 4.26	-0.34 | -1.14
With Sinusoidal (Lat/Lon)	0.37 | 0.58	6.11 | 4.33	-2.19 | -0.87
GeoCLIP Location Embeddings	0.50 | 0.63	5.40 | 4.09	-0.03 | 0.26