Authors' Response to Reviews of

A Transformer-based agent model of GEOS-Chem v14.2.2 for informative prediction of $PM_{2.5}$ and O_3 levels to future emission scenarios: TGEOS v1.0

Dehao Li, Jianbing Jin*, Guoqiang Wang, Mijie Pang, Hong Liao*

Geoscientific Model Development Discussions, 10.5194/egusphere-2025-2186

RC: ***Reviewers' Comment***, AR: Authors' Response, □ Manuscript Text

1. Overview

Response to Referee 1: We would like to thank the referee for the careful review throughout the paper and the in-depth comments that help to improve our paper.

2. Major concerns

RC: ***My primary concern lies in the lack of appropriate model comparisons and insufficient clarity regarding model performance. Specifically, I would like to understand why the authors chose to compare their model against Multi-Layer Perceptron (MLP) and Random Forest (RF), rather than Convolutional Neural Networks (CNNs), given that they mentioned DeepRSM (Xing et al., 2020) earlier in the text but did not include it in their evaluation. Although the authors state that a series of (hyper)parameter tuning experiments were conducted (L407), it remains unclear how these were performed. For example, using 300 trees and a maximum depth of 25 in the RF model can easily lead to overfitting. This raises the question of whether these baseline models were properly tuned, which may have contributed to their underperformance. Furthermore, while the authors report low RMSE and MAE of the TGEOS v1.0, it is unclear what constitutes "low" in this context. Additional comparisons with values reported in other relevant studies would strengthen the claims of model performance.***

AR: Thank you very much for your thoughtful comment regarding model comparisons and performance in this manuscript. First and foremost, we would like to clarity the model comparison between CNN and our TGEOS. In this study, we did not select the CNN architecture employed in DeepRSM (Xing et al., 2020) for comparison due to the distinct form of the input data derived from TGEOS. The input for our model primarily consists of sequential samples from individual grid points, which is not well-suited for CNNs to process effectively, as will be specifically discussed in our response to Point 3. After considering the reviewer's suggestions, we attempted to construct a CNN-based model to predict same target variables as TGEOS for model comparison. The architectural overview of this model is illustrated in Figure 1. In this CNN-based model, we transformed the feature input of each sample from its original dimension of (1, 1045) into a matrix format of (9, 116). For the temporal features (i.e., months corresponding to each scenario in this study), we individually convert them into embedding vectors—following an approach commonly used in NLP—and subsequently concatenate these vectors with the flattened output of the final convolutional layer of the CNN before feeding them into the fully connected layer. We optimized the hyperparameters using Optuna and conducted tuning experiments based on seven key parameters: "Batch size", "Learning rate", "Epoch", "Kernel size", "Padding", "Number of channels 2 (number of second convolution channel)" and "Size of full connect 1 (dim of first FC output)", with R² and MAE selected as the optimization objectives. To reduce computational complexity, 40% of the dataset was randomly sampled for training in each epoch, while an additional 10% was reserved for validation purposes. The detailed experimental options are presented in Table 8.

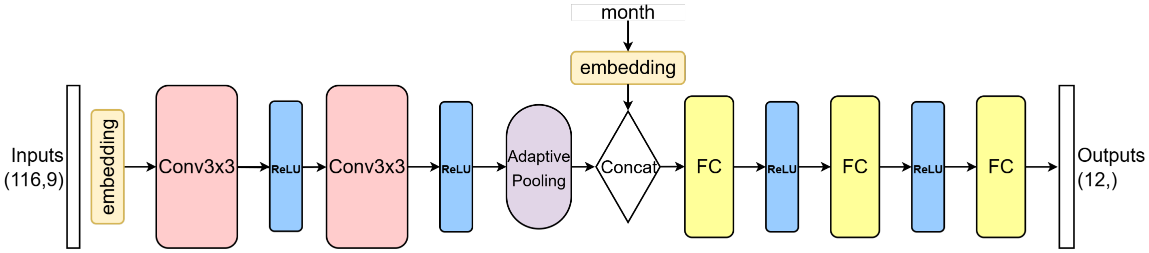

Figure 1: Model architecture of CNN model.

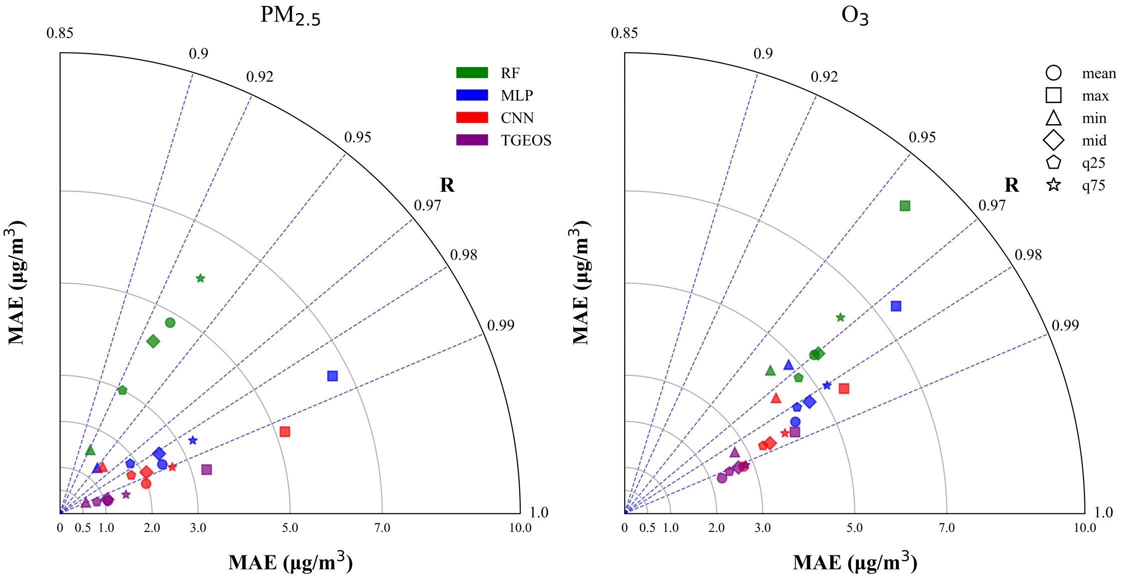

Figure 2: Predictive performance of four models, with green represents RF predictions, blue denotes MLP predictions, red denotes CNN predictions, and purple indicates TGEOS predictions. All indicators are averaged in national scale and computed based on the six test scenarios.

Based on the aforementioned tuning parameters, we utilized the same training set as TGEOS for model training and evaluated the model's performance using the identical test set. As illustrated in Figure 2, the performance of the four models on the test set is compared. On one hand, compared to the previously selected MLP and RF models, the CNN-based model demonstrates superior performance, characterized by higher R values as well as lower MAE. This advantage can be attributed to the CNN's local convolution kernel, which is capable of capturing patterns among adjacent data points. On the other hand, when compared to TGEOS employed in this study, the CNN-based model underperforms across all evaluation metrics. This is primarily due to TGEOS's self-attention mechanism, which enables more effective dynamic and global modeling. In contrast to CNNs, which are constrained by fixed convolution kernels and limited network depth, Transformer-based TGEOS exhibits a stronger capacity for capturing complex relationships in high-dimensional data.

In addition, we appreciate the reviewer's concern regarding the hyperparameter tuning process. To ensure fair and robust comparisons, we used Optuna to perform automated hyperparameter tuning for four models, including RF, MLP, CNN, and our proposed TGEOS model. The tuning results are presented in Fig .4 to 8. For each model, we defined a relevant hyperparameter search space. Except for CNN model aforementioned, in the RF model, we tuned the number of trees (n_estimators), maximum tree depth, and minimum samples per leaf. In the MLP, we tuned the number of layers, hidden units, activation functions, learning rate, and dropout rate. For each tuning epoch, 40% of the training set was randomly sampled for training, with additional 10% was for validation. The test set was strictly reserved for final evaluation. Through this tuning process, we ensured that each model was evaluated using its optimized configuration, thereby reducing the risk of unfair comparisons due to under-tuned baselines. We will clarify these details in the revised manuscript and include the hyperparameter search spaces in the supplementary material.

Regarding the reviewers' concerns about the RF model, here we present the top five configurations of hyperparameters with the highest scores in the tuning experiment. As shown in Table 1, the performance of the RF model remains highly consistent across different hyperparameter configurations. The average R^2 values fluctuate only slightly between 0.74, while the corresponding MAE values range from 5.100 g/m^3. This indicates that the model performance is relatively insensitive to variations in these hyperparameters once the model reaches a sufficient level of complexity. In other words, the this model already possesses adequate capacity to capture the underlying relationships in the dataset.

Table 1: Five hyperparameter combinations with best R^2 and MAE.

N estimators	Max depth	Min samples split	Min samples leaf	Avg R^2	Avg MAE
300	25	4	2	0.7404	5.059
200	50	5	10	0.7401	5.060
100	25	10	9	0.7382	5.070
100	15	6	6	0.7363	5.104
300	15	7	4	0.7336	5.116

Finally, we appreciate the reviewer's interest in benchmarking against other emulating studies such as RSM and NN-CTM, as TGEOS differs from these models in terms of time resolution, learning objectives, and applicable scenarios, making direct comparison infeasible. For example, DeepRSM uses CMAQ as its target and is designed specifically for response prediction under a uniform regional emission coefficient (Xing et al., 2020), limiting its applicability to more detailed emission scenarios like DPEC-SSP/DPEC-CA scenarios used in this study. Therefore, we did not consider it an appropriate direct baseline for comparison. Alternatively, we compared TGEOS against several widely used machine learning models (MLP, RF and CNN) under the same future scenario simulation framework to evaluate predictive skill of TGEOS.

While DeepRSM was trained based on a set of Latin Hypercube Sampled (LHS) emission perturbation scenarios that gives the model some capacity to generalize across emission changes, the evaluation of

DeepRSM was conducted on scenarios based on a specific historical year (2017), and the model was applied primarily to reproduce air quality responses within that known temporal context. This differs from our goal, where TGEOS is designed to emulate GEOS-Chem outputs under projected future emission scenarios. For another thing, most of previous studies were developed specifically for the CMAQ model, while our TGEOS model emulates outputs from the GEOS-Chem chemical transport model. These two CTMs differ in their chemical mechanisms, spatial resolutions, input structures, and internal processing. Therefore, the input–output relationships learned by these models are not directly transferable or comparable to those learned by TGEOS. For these reasons, although we cited these papers to highlight prior efforts concerning the CTM emulating works, we did not consider it an appropriate direct baseline for comparison. Alternatively, we compared TGEOS against several widely used machine learning models (MLP, RF and CNN) under the same future scenario simulation framework to evaluate predictive skill and computational efficiency.

To validate the performance of the TGEOS model in "emission-concentration" modeling against other machine learning models, ~~two widely used machine learning models, including multilayer perceptrons (MLP) and random forests (RF)~~ three widely used machine learning frameworks, namely Multilayer Perceptrons (MLP), Random Forests (RF), and Convolutional Neural Network (CNN) employed in previous studies (Xing et al., 2020; Xing et al., 2021), were simultaneously employed based on the multi-scenario dataset mentioned in Section 2.1. For each ML model, we identified the model ~~that demonstrated optimal fitting performance for testing after conducting a series of parameter tuning experiments.~~ with the best combination of hyperparameters after fine-tuning process based on Optuna tool. The MLP model uses 4 hidden layers with 2048, 1024, 512, and 256 neurons, applying ReLU activation and Dropout to prevent overfitting. The optimizer is Adam with a learning rate of $1e^{-3}$, and the loss function is Mean Squared Error (MSE). Training uses a batch size of 1024 and 100 epochs, with a learning rate scheduler to adjust the learning rate dynamically. The RF model uses 300 trees with a maximum depth of 25, a minimum sample split of 4, and a minimum sample per leaf of 2. It uses parallel computation with all CPU cores and performs feature selection by choosing the top 500 important features. The model consists of two convolutional layers followed by fully connected layers, with an additional embedding layer to incorporate month information. The first convolutional layer applies 32 filters of size 3×3 (padding = 1) to the single-channel input, followed by a second convolutional layer with 128 filters of the same size. Both convolutional layers use ReLU activations. The output feature maps are then processed by an adaptive average pooling layer to reduce the spatial resolution to 29×3. To integrate temporal information, a month embedding layer maps month indices (1–12) to a 4-dimensional vector. The pooled convolutional features are flattened and concatenated with the month embedding, forming the input to a three-layer fully connected network: the first linear layer maps the concatenated vector to 256 units, the second reduces it to 64 units, and the final output layer produces 12 regression targets. ReLU activation functions are applied after the first and second fully connected layers. For each model, hyperparameters were obtained after fine-tuning based on Optuna tool.

Table S2 and S3 summarize the performance of the three models on the test set. We found that TGEOS outperforms the other two models in both R^2 and MAE metrics. To clearly illustrate the predictive performance of different models, we presented a modified Taylor diagram (Taylor, 2005; Fang et al., 2023) in Fig. 11. This diagram simultaneously displays the Mean Absolute Error (MAE) and correlation coefficient (R) for predictions of PM$_{2.5}$ and O$_3$ indicators from three models in various regions. Our findings indicate that the Random Forest (RF) model performs the poorest. This is primarily due to its reliance on feature importance assessments during feature selection, which overlooks potential underlying features in the data, adversely affecting the model's fitting capability. Additionally, the

> RF model is sensitive to the distribution of training data, leading to limited extrapolation abilities and poor predictive performance for extreme values. In contrast, the Multi-Layer Perceptron (MLP) shows a significant improvement in predictive performance relative to the RF model. Leveraging its multi-layer neural network structure, the MLP can more effectively learn complex relationships between multiple features. But this layered structure can struggle when dealing with high-dimensional feature spaces, especially for highly stochastic indicators such as maximum values, where the MLP still exhibits considerable prediction errors. Compared to the previously selected MLP and RF models, the CNN-based model demonstrates superior performance, characterized by higher R values as well as lower MAE. This advantage can be attributed to the CNN's local convolution kernel, which is capable of capturing patterns among adjacent data points.
>
> Conversely, the Transformer-based TGEOS model demonstrates superior performance compared to the other models, exhibiting higher R values (exceeding 0.98 and 0.97) and lower MAE values (less than 4.0 g/m^3 for the majority indicators). These results suggest a higher degree of reliability and accuracy in its predictions. For several indicators where MLP performs poorly, TGEOS demonstrates substantial improvements. The superiority of the Transformer model can be attributed to its greater number of parameters and more complex architecture, which leverage powerful feature extraction capabilities and self-attention mechanisms, allowing it to adapt to intricate patterns and relationships. In contrast to CNN, which are constrained by fixed convolution kernels and limited network depth, TGEOS exhibits a stronger capacity for capturing complex relationships in high-dimensional data. ~~Consequently, in high-dimensional tasks like air quality modeling, Transformer models have proven to be more advantageous compared to their counterparts.~~

RC: ***I am also concerned about the authors' treatment and definition of RSMs. In L63–68, RSMs are introduced as a statistical method for "emission-concentration" estimates, but later DeepRSM is mentioned, and then the authors refer to machine learning being effective for "emission-concentration" modeling without relying on RSM (L111-114). It would be helpful if the authors could clarify what exactly qualifies as an RSM in this context. Does a model need to be derived from CTM outputs to be considered an RSM? And if the goal is simply to emulate CTM outputs, wouldn't it be more appropriate to use the broader and more inclusive term "emulator"?***

AR: Thank you very much for your thoughtful comment regarding the treatment and definition of RSMs in our manuscript. We appreciate the opportunity to clarify this important point.

In this study, we adopt the term "emulator" or "agent model" in a broad sense to denote surrogate models that approximate the outputs of chemical transport models (CTMs) using key CTM inputs, such as emissions and meteorological conditions. Both Response Surface Models (RSMs, e.g., DeepRSM) and the proposed TGEOS model fall into this category of emulators. For one thing, the TGEOS model is built upon key GEOS-Chem inputs (mainly concerning emission variations) and pollutant concentrations, and is specifically designed for rapid prediction of air pollutants concentrations aligned with GEOS-Chem simulations under future emission scenarios. For another thing, RSMs were constructed based on the nonlinear relationship between emissions and concentrations using statistical methods, enabling rapid estimation of pollutant concentrations under varying emission scenarios. This characteristic makes RSMs closely aligned with the TGEOS model used in this study at the application level, leading us to focus primarily on the limitations of RSMs.

The construction of RSMs requires a large number of CTM simulations, resulting in substantial computational costs, particularly when extended to multiple pollutants, precursors, and large spatial domains (Zhao et al., 2015). To address this challenge, DeepRSM leverages machine learning techniques to streamline the RSM framework. Its demonstrated success highlights the strength of ML in handling high-dimensional air quality modeling tasks. Building on this insight, several studies have developed CTM emulators based entirely on machine learning (Huang et al., 2021; Liu et al., 2022), without reliance on traditional statistical structures like RSMs, to directly predict pollutant concentrations. This illustrates the greater flexibility and efficiency of ML-based approaches in emission–concentration emulator modeling. In other words, the statement in our manuscript, “showcasing the efficacy of machine learning in "emission-concentration" modeling without relying on RSM”, is intended to highlight the advantages of machine learning in high-dimensional air quality modeling, and to emphasize its potential as an optimized alternative to conventional RSM methods, capable of independently performing rapid “emission–concentration” calculations.

We have revised the description of the corresponding part in the manuscript, with details shown in blew.

1 Introduction (L59-L106)

~~To address the computational challenge and efficiently retrieve the nonlinear relationship between emissions and concentrations, data-driven statistical emulators have been proposed to accelerate numerical simulations (Castruccio et al., 2014). A reliable emulator can accurately depict intricate relationships between inputs and outputs, such as from emissions to concentrations. It can also faithfully approximate the fundamental mechanisms of atmospheric models, thereby generating numerical simulations that exhibit a high degree of consistency to the model (Salman et al., 2024). Among all the emulators, Response Surface Model (RSM) is the most widely used method. It is a statistical method developed by the US EPA (EPA, 2006) that uses the maximum likelihood estimation - empirical best linear unbiased predictors (MLE-EBLUPs) technique (Santner et al., 2003) to establish the complex relationships between emission rates of several pollutants and the responses they produce on the pollutant concentrations by fitting response surfaces of the nonlinear system (Box and Draper, 2007), and provide best estimate of the pollutant. When given some unknown emission scenarios, RSM can rapidly retrieve the changes of aimed concentrations without additional CTM simulation involved (Wang et al., 2011). RSM technique has been successfully employed in the response modeling of $PM_{2.5}$ (Wang et al., 2011) and ozone (Xing et al., 2011) to precursor emissions in China for typical regions. Since conventional RSM commonly requires a large number of CTM simulations to fit reliable response surfaces (Xing et al., 2011; Zhao et al., 2015), notable advances focusing on enhancements in both efficiency and accuracy in RSM technology have been achieved (Li et al., 2022). For example, Extended Response Surface Models (ERSMs) (Zhao et al., 2015; Xing et al., 2017) allow for the incorporation of a greater number of variables and geographical regions, improving alignment with independent CTM simulations compared with traditional RSM (Zhao et al., 2015; Xing et al., 2017). Moreover, the polynomial function based RSM (pf-RSM) is capable of quantifying the nonlinear relationships between air pollutant concentrations and precursor emissions by fitting CTM simulations to a series of polynomial functions and mitigating the computational burden through decreasing the number of required CTMs up to 60% (Xing et al., 2018). Recently, many studies have used novel machine learning techniques to accelerate the modeling process of RSM by further reducing the number of required CTMs. For instance, Deep-RSM, developed by Xing et al. (2020) using convolution neural networks (CNN), requires only two CTM cases (i.e., base and control scenarios) to startup the model; Self-adaptive RSM (SA-RSM, Li et al. (2022)) further reduces the number of required CTMs for pf-RSM modeling by employing a stepwise regression method to estimate the coefficients of polynomial functions.~~

~~Although existing RSM techniques exhibit more efficiency than traditional CTM in predicting the response of pollutant concentrations to a wide range of emission changes, there are still several issues to be addressed. Firstly, due to the structural limitations that restrict the model from executing~~

~~multi-target predictions, existing techniques focus mainly on the response of average of the target pollutants over a period of time, such as the monthly average (Huang et al., 2021). However, predicting the singular monthly average of pollutant concentrations may overlook critical variations throughout the month, such as extreme values (Guo et al., 2020; Zhao et al., 2022). Therefore, these approaches fall short in providing a comprehensive evaluation of future pollution states, including the ability to identify potential extreme pollution events under various emission scenarios. Secondly, RSM techniques rely on the polynomial assumption, leading to its disadvantage to cope with high-dimension problems. As the number of input variables increases, the complexity of RSM model grows, necessitating a larger number of samples for accurate fitting (Zhao et al., 2015) and potentially leading to multi-collinearity issues (Xing et al., 2018). This limitation restricts the applicability of RSM to more intricate emission scenarios. Therefore, existing RSM studies have primarily concentrated on emissions of a few major pollutants and the add-up emissions, failing to address air quality response under more detailed scenarios that incorporate sectoral emissions and a broader range of emission species. While ERSM considers emission sectors (Zhao et al., 2015), the inherent limitations of RSM in handling high-dimensional data result in a substantial requirement for CTM samples, thus confining its application to modeling studies in smaller areas. Thirdly, current RSMs e.g. pf-RSM (Xing et al., 2018) and SA-RSM (Li et al., 2022) account for each spatial grid independently while neglect the impact of surrounding emissions, which have been shown to affect local pollutant concentrations (Cheng et al., 2019). While ERSM (Zhao et al., 2015) has considered regional transport of emissions, it requires a substantial number of scenario simulations to ensure the accuracy of the model (Zhao et al., 2015; Xing et al., 2017). For example, modeling for a middle-scale region typically necessitates hundreds of scenarios as support (Zhao et al., 2015). The computational burden significantly limits the application of this technology on a national scale. In summary, given that existing techniques inadequately address the challenges associated with high temporal-resolution prediction, inapplicability of multivariate scenarios, and negligence of emission transport, developing a comprehensive national-level "emission-concentration" predictive model poses a significant challenge.~~

To overcome the computational challenge and efficiently retrieve the nonlinear relationship between emissions and concentrations, data-driven statistical emulators have been proposed to accelerate numerical simulations (Castruccio et al., 2014). As a simplified-form of CTM, a reliable emulator can effectively capture the intricate relationships between important CTM inputs and concentration outputs, and rapidly estimate "CTM-aligned" concentrations of pollutants. Response Surface Model (RSM), served as statistical surrogates developed by the US EPA (EPA, 2006) to establish the relationships between emission rates and the concentration responses of CTM, has been continuously developed since the past decade. RSM techniques have been successfully employed in the response modeling of $PM_{2.5}$ (Wang et al., 2011) and O_3 (Xing et al., 2011) to precursor emissions in China for typical regions. To address the inherent computational burden stemmed from considerable advanced CTM supports for model building (Xing et al., 2011), optimized versions of conventional RSM were developed, such as ERSM (Zhao et al., 2015; Xing et al., 2017) and pf-RSM (Xing et al., 2018). Recently, novel machine learning (ML) techniques, for its well performance in simulating complex non-linear relationships in atmospheric systems (Liu et al., 2021) and dealing with tasks involving multiple variables and objectives (Masmoudi et al., 2020; Huang et al., 2021), have been employed in RSM techniques to further optimize modeling efficiency and estimation accuracy of RSMs (Xing et al., 2020; Li et al., 2022). Based on this advantage, many studies have attempted to build effective emulators using pure ML method (Huang et al., 2021; Zhang et al., 2023a). For example, Zhang et al. (2023a) used ResCNN framework to predict annual $PM_{2.5}$ concentration from fossil energy use and reveal the co-benefits of the energy transition, demonstrating the potential of ML method in addressing the emulator modeling task.

Although existing CTM emulators exhibit more efficiency than traditional CTM in estimating the pollutant concentrations to a wide range of emission changes, there are still several issues to be addressed. Firstly, due to the computing limitations (Liu et al., 2022), the temporal resolution for some emulators was constrained with annual scale, which greatly prevent these emulators from providing detailed estimations of air pollutants such as extreme values throughout the year (Guo et al., 2020; Zhao et al., 2022). Secondly, while some emulators have the ability to offer concentration estimations with finer temporal resolution, they still have limitations. On one hand, RSM-based emulators rely on the polynomial assumption, leading to its disadvantage to cope with high-dimension problems. As the number of input variables increases, the complexity of RSM model grows, necessitating a larger number of samples for accurate fitting (Zhao et al., 2015) and potentially leading to multi-collinearity issues (Xing et al., 2018). In the revised manuscript, we will provide examples (BTH, YRD) to avoid ambiguity. This limitation restricts the applicability of RSM-based emulators to more intricate emission scenarios. Therefore, existing RSM studies have primarily concentrated on emissions of a few major pollutants and the add-up emissions (Xing et al., 2020), failing to address air quality response under more detailed scenarios that incorporate sectoral emissions and a broader range of emission species. On the other hand, some emulators were constructed based on in-situ observations using ML method (Zhang et al., 2023a), which is easy to employ and more convenient than those RSM-based emulators. However, these models are constrained by the limited number of observational data stations and are therefore unable to effectively assess air quality in regions where observational infrastructure is lacking (Xu et al., 2022). Furthermore, due to insufficient observational data, these models often do not have enough representative samples to achieve accurate model fitting, which leads to suboptimal predictive performance (Tang et al., 2024). In addition, traditional ML models, such as Multi-Layer Perceptron (MLP) and Random Forest (RF), may not fully capture the nonlinear relationships in complex atmospheric variables (Masmoudi et al., 2020; Natarajan et al., 2024; Abuouelezz et al., 2025), which further undermine their predictions. Thirdly, some current emulators account for each spatial grid or observation site independently while neglect the impact of surrounding emissions (Xing et al., 2018; Li et al., 2022; Zhang et al., 2023a), which have been shown to affect local pollutant concentrations (Cheng et al., 2019). Although certain studies have employed convolutional neural network (CNN) architectures capable of capturing local features to develop models (Xing et al., 2020; Huang et al., 2021; Liu et al., 2022), the computational resource constraints have hindered these "face-to-face" models from processing large volumes of feature inputs. As a result, the application of such models is limited in terms of emission details and research domain. In summary, given that existing techniques inadequately address the challenges associated with high temporal-resolution prediction, inapplicability of multivariate scenarios, and negligence of emission transport, it still be a significant challenge to develop a comprehensive emulator using more advanced method.

RC: ***The manuscript does not sufficiently demonstrate the advantages of using the Transformer architecture, nor does it clearly describe the model's structure. Beyond the comparative limitations mentioned in point 1, the authors do not provide any analysis of computational complexity between Transformers and CNNs. Instead, they state that CNN "may increase the demand for computational resources, especially when addressing considerable features" (Lines 208–219), without supporting this claim with the data. Additionally, the input to a Transformer is typically a sequence of tokens, yet the authors refer to them as "channels," which may cause confusion. In image-based applications, Vision Transformers (ViTs) have shown strong performance, and it is unclear why the authors didn't explore spatial structures through CNNs or ViTs.***

AR: We appreciate the reviewer's concern regarding the rationale for using a Transformer architecture. First and foremost, upon reviewing Lines 208–219 of the manuscript, we would like to acknowledge that our original wording may have caused confusion, and we appreciate the opportunity to clarify. In this section, our statement about increased computational cost was not intended to refer to CNNs in general, but rather to CNN-based field-to-field modeling approaches, such as those used in (Xing et al., 2020; Huang et al., 2021; Liu et al., 2022). In their work, both the input and output are represented as high-resolution 3D matrices (spatial fields), which require significant GPU memory and computational resources—especially when the number of input variables increases. As a result, they limited their model to only a few types of emission fields as inputs (Xing et al., 2020), or solely average predictions (Liu et al., 2022). In contrast, our dataset includes over 100 variables, including sectoral emissions and multiple meteorological parameters. Representing these as full spatial fields and training a CNN-based model in a field-to-field form was not feasible under our available computational resources. Therefore, instead of modeling spatial fields directly, we adopted a high-dimensional sequential modeling strategy. Our dataset is not field-based but rather consists of structured multivariate sequences, in which spatial and feature-level information (e.g., emissions, meteorology, and concentrations at multiple grid points) is flattened and treated as a sequence of tokens fed into TGEOS model. This approach offers a more scalable solution while preserving the ability to capture complex relationships among variables across grid points.

Although the input features are derived from multiple spatial grid points (e.g., a central location and its adjacent neighbors), they are organized as flattened feature vectors rather than structured fields. This flattening also removes the explicit spatial topology (e.g., 2D grid layout) that is critical to models like CNNs and ViTs commonly used for field-based image data (Li et al., 2021). In contrast, the Transformer model was originally designed for sequential data (Vaswani, 2017) and has since shown great promise in multivariate time series modeling (Li and Moura, 2019; Zerveas et al., 2021). Compared to CNNs or MLPs, Transformers are better suited for capturing long-range dependencies, complex inter-feature relationships, and global patterns without the limitations of fixed local receptive fields (Zhao et al., 2021; Khan et al., 2022). As a result, we chose to use the Transformer architecture to build the model for its flexibility and its proven effectiveness in modeling complex dependencies in high-dimensional, structured input data. The Since our data do not possess spatial locality in the image sense, but rather form feature sequences across domains. The explicit model's structure of TGEOS is presented blew.

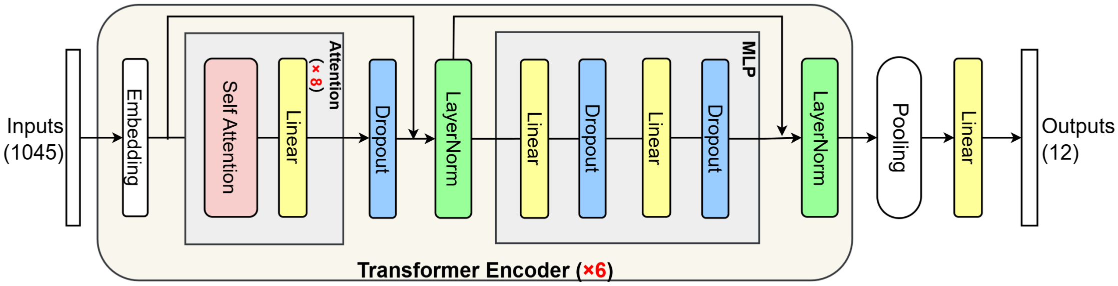

Figure 3: Model architecture of TGEOS v1.0.

To further elucidate the model structure, we conducted a comparative analysis of the computational complexity across different models employed in this study, focusing on three key metrics: the number of parameters, inference speed, and memory usage. It is well-documented that Transformer architectures generally impose a higher computational cost compared to other model types. As evidenced by multiple studies (e.g., Dosovitskiy et al. (2020);Wu et al. (2021);Zerveas et al. (2021)), Transformers typically require more parameters and consume greater memory due to the quadratic complexity inherent in the self-attention mechanism. This characteristic often leads to extended training durations and increased GPU memory utilization, particularly when processing long input sequences or high-dimensional features. Nevertheless, Transformers offer substantial benefits in capturing global dependencies and modeling intricate feature interactions—capabilities that are essential for our task, which involves structured multivariate data with implicit spatial relationships.

Table 2: Comparison of complexity of the four models.

Model	Total parameters	Training time	Inference time
RF	-	181.6 min	1.24s
MLP	4.91 M	144.1 min	0.83s
CNN	2.91 M	161.2 min	3.99s
TGEOS	22.66 M	192.2 min	1.26s

In addition, we also thank the reviewer for pointing out the potential confusion in our terminology. We agree that the term “channel” may be misleading in the context of Transformer architectures, especially since “channels” are more commonly used in CNNs to describe the depth dimension of image-like inputs. In our work, each input sample is represented as a high-dimensional feature vector consisting of variables from multiple grid points or adjacent spatial locations. When reshaped for input into the Transformer, this feature vector is treated as a sequence of tokens, where each “token” corresponds to a distinct spatial or physical feature (e.g., emissions, meteorological variables at a specific grid or neighbor location). Thus, the input to the Transformer is a 2D tensor of shape (sequence length, embedding dimension), consistent with standard Transformer input conventions in NLP and time-series modeling. The term “channel” was originally used in an informal sense to refer to different input components, but we acknowledge that it may cause confusion and will revise the manuscript to adopt more accurate terminology, such as “tokens”, “input embeddings”, or “feature sequences”, to align with Transformer literature and avoid misinterpretation.

We have revised our manuscript in accordance with the aforementioned discussions, correcting these inappropriate statements and clarifying the advantages of the Transformer architecture compared to other models in order to justify our architectural choice.

2.2.1 Model architecture (L206-L227)

In previous "emissions-concentration" modeling, field-to-field modeling using the convolution neural networks (CNN) model has been widely used because of the efficient usage of the spatial relationship between features and concentrations (Xing et al., 2020; Huang et al., 2021). ~~However, this approach may increase the demand for computational resources, especially when addressing considerable features.~~ In their study, both inputs and outputs were represented as high-resolution three-dimensional spatial fields, which demands substantial GPU memory and computational power—particularly as the number of input variables grows. Consequently, their model was restricted to only five types of emission fields, without the capacity to incorporate a broader range of pollutant species or more finely resolved sectoral emissions. By contrast, our dataset comprises more than one hundred variables, including sector-specific emissions and a wide range of meteorological parameters. Representing all of these as full spatial fields and applying a CNN in a field-to-field manner would have exceeded our available computational capacity. To address this, we adopted a high-dimensional sequential modeling framework. Rather than using field-based representations, our dataset is organized as structured multivariate sequences, where spatial and feature-level information (e.g., emissions, meteorology, and pollutant concentrations at multiple grid points) is flattened into a sequence of tokens for input into the TGEOS model. This design enables scalability while maintaining the ability to model complex inter-variable relationships across spatial locations.

In this study, we employed an informative prediction model based on Transformer architecture comprising the encoder for feature extraction and the regressor for target mapping. In order to align with the shape of the dataset, ~~the model was designed with 1048 input channels and 12 output channels~~ the model was configured with an input feature dimension of 1045 and an output dimension of 12. . Six Encoder layers were configured with the model, each of which primarily incorporates a multi-head self-attention mechanism with eight attention heads and a feed-forward network. The multi-head self-attention mechanism was employed to capture the dependency relationships among various positions within the input sequence, while the feed-forward network facilitates additional nonlinear transformations on the features at each position (Vaswani, 2017). By leveraging the multi-head self-attention mechanism, the model can compute the similarity (or attention weights) of each feature in relation to all other features, thus producing a weighted representation for each position and determining the extent to which each position relies on information from others. Moreover, the feed-forward network, consisting of two fully connected layers, enhanced feature representation and improves the model's learning efficacy by incorporating nonlinear activation functions. In this implementation, the ReLU activation function was selected due to its ability to prevent negative values and expedite the model's training process (Nair and Hinton, 2010). Additionally, each sub-module incorporated residual connections and layer normalization to mitigate the risks of gradient disappearance or explosion. The output from the Encoder undergoes global pooling to decrease model complexity. Finally, the regressor was comprised of fully connected layers that map the Encoder output to the specified output ~~channels~~ sequences .

3. Minor concerns

RC: ***L81 ("to startup the model"): "Startup" is not typically used as a verb in this context.***

AR: Thank you for pointing this out. We have revised this part and will pay attention to the use of "startup" in the future.

RC: ***L167-168 ("fine-tuning experiments"): I understand that the authors are referring to "fine-tuning experiments" in the context of data assimilation (Text S1), but in a machine learning/AI context, the term "fine-tuning" is typically associated with pretraining followed by fine-tuning, which may cause confusion for readers.***

AR: Thank you for your insightful comment. We agree that the term "fine-tuning" may be misleading in the context of machine learning, where it typically refers to adapting a pretrained model to a specific task. In our manuscript, the intended meaning was closer to "perturbation scenarios using data assimilation tuning method". To avoid confusion, we have revised the term to "perturbation scenarios" in Line 167–168 and clarified the description in Text S1 accordingly.

2.1.1 Multi-scenario inventory (L167-L172)

In addition, in order to improve the generalization ability of the model, we designed ~~random scenarios~~

~~based on fine-tuning experiments~~ 11 perturbation scenarios using data assimilation tuning method (denoted as "Tuning scenarios" in Table 1) , including emission scenarios with different emission factors ranging from 0 to 2.0 for each emission species and emission sector, ~~so that the model can better understand the relationships between various features and model performance can be substantially improved.~~ thereby expanding coverage of the input space and reducing the risk of extrapolation to unseen values, especially for those predictions under high emission scenarios. These emission factors were generated for representing the spatial variability that widely used in data assimilation (Jin et al., 2023). The detailed process for generating these stochastic emission factors is discussed in the Text 1.

RC: ***L175-176 ("we set the maximum value of each coefficient matrix to 2.0"): Any justifications?***

AR: Thank you for this comment. We set this value to 2.0 because we investigated it the raw ratio matrices in advance, and found that over 80% of grids had coefficients below 2.0, with most values in key emission regions concentrated between 0.3 and 1.5. Coefficients exceeding 2.0 occurred almost exclusively in very low-emission grid cells (typically < 0.1 ton per grid and located in western China), where differences between DPEC and MEIC inventories in absolute terms are small but amplified in ratio form. Although some previous studies have adopted lower thresholds (e.g., 1.2 for (Xing et al., 2011)), other research suggests that future emissions in certain heavily polluted regions could plausibly exceed 1.5 Brean et al. (2023), making 2.0 a more flexible yet still reasonable threshold.

2.1.1 Multi-scenario inventory (L173-L179)

~~For each emission scenario, we divided it by the 2017 MEIC inventory to obtain a series of monthly coefficient matrices for the emissions of various species in five sectors. It is worth noting that since the units of DEPC and MEIC data are tons per grid, significant variation are exhibited between adjacent grids. Thus, we set the maximum value of each coefficient matrix to 2.0 when making the DPEC scenarios to avoid abnormal emission coefficients due to magnitude differences. Subsequently, we multiplied the generated coefficient matrices to the corresponding part of the original inventory used for GEOS-Chem input to obtain the multi-scenario inventory that reflect the control of each scenario. The new inventory was employed in GEOS-Chem simulating to obtain~~ ~~$PM_{2.5}$ and O_3~~ ~~concentrations under future emission scenarios.~~
Since the unit of DPEC-SSP/CA emissions is t/grid, which is incompatible for GEOS-Chem running, we used MEIC inventory with t/grid as unit at 2017 as a benchmark (denoted as b-MEIC), and make elementwise divisions between DPEC and b-MEIC to obtain a series of monthly emission coefficient matrices for various species in five sectors, namely power, industry, residential, transportation, and agriculture. Since the majority of grids with emission factor smaller than 2.0 (>80%) and to prevent abnormal values due to magnitude difference of two inventories, the threshold of emission factors was artificially set to 2.0. Subsequently, we took the Schur product of the coefficient matrices and corresponding part of MEIC inventory used for GEOS-Chem input, with unit of kg/m²/s, to generate emission inventories projected with DPEC-SSP/CA.

RC: ***L181 ("GEOS-Chem chemistry transport model"): It should be chemical transport model (based on the definition https://geoschem.github.io/overview.html).***

AR: Thank you for pointing this out. We have corrected "chemistry transport model" to "chemical transport model" in Line 181 to align with the official definition from the GEOS-Chem documentation.

2.1.2 GEOS-Chem configuration (L181)

The GEOS-Chem ~~chemistry~~ chemical transport model (http://www.geos-chem.org, version 14.2.2) was used to simulate the spatiotemporal distribution of surface $PM_{2.5}$ and O_3 concentrations under different emission scenarios based on year 2017.

RC: ***L195 ("8 key meteorological parameters"): In Table 2, it says "(2) 9 meteorological parameters."***

AR: Thank you for pointing out this inconsistency. There are actually 9 meteorological parameters mentioned in this article, namely 2-meter air temperature (T2M), 10-meter northward wind (V10M), 10-meter eastward wind (U10M), planetary boundary layer height (PBLH), 2-meter specific humidity (QV2M), total precipitation (PRECTOT), relative humidity (RH), evaporation from turbulence (EVAP), and surface pressure (PS). We have corrected the text in Line 195 to match Table 3, now referring to "9 meteorological parameters" for consistency and accuracy, and supplement some details.

Table 3: Targets and features for TGEOS model.

Target number	Training targets	Feature number	Training Features
12	Monthly average, maximum, minimum, median, 25 and 75 percentiles of $PM_{2.5}$ and O_3 concentrations	1045	(1) power, industry, residential, transportation, and agriculture emissions: NH3, PM2.5, OC, PM10, BC, CO, NO, SO2, RCHO, XYLE, ALK2, CCHO, OLE2, ALK5, HCHO, TOLU, ALK4, ALK3, EOH, ETHE, MOH, ALK1, MEK, OLE1, ACET, MACR, as well as 8 adjacent sectoral emissions. (2) 2-meter air temperature (T2M), 10-meter northward wind (V10M), 10-meter eastward wind (U10M), planetary boundary layer height (PBLH), 2-meter specific humidity (QV2M), total precipitation (PRECTOT), relative humidity (RH), evaporation from turbulence (EVAP), and surface pressure (PS), as well as 8 adjacent meteorology. (3) local and adjacent longitude and latitude values, and month in each scenario.

RC: ***L213 ("an informative prediction model"): I am not entirely sure how the authors define the term "informative," which appears multiple times throughout the manuscript, including in the title and abstract. It would be helpful to clarify what is meant by "informative" in this context.***

AR: Thank you for pointing this out. In this context, we used the term "informative" to convey that the TGEOS v1.0 model not only provides average values as other models do (Liu et al., 2022), but also predict multiple statistical concentration indicators (e.g., 75-percentile and max values). We agree that the term may be vague. To improve clarity, we have clarified the meaning of "informative" in the introduction part of the manuscript.

1 Introduction (L127-L143)

~~In this study, we proposed an highly-efficient and informative air quality agent prediction model (TGEOS v1.0), which is based on Transformer architecture as a GEOS-Chem agent model. It is referred to as "TGEOS" throughout this paper. Relative to earlier studies, this research offers several significant benefits. First, TGEOS is able to predict the probability distribution of future air quality under different emission scenarios. Compared to solely average estimated by previous RSM methods, probability distribution can provide informative frequency distributions of pollutants (Yang and Wu, 2022). The probability distributions have been used to represent future states of $PM_{2.5}$ (Li et al., 2024) and O_3 (Zeng et al., 2022) concentrations in diverse emission scenarios, and to explore any extreme pollution events that are typically represented by the high-end tail of the probability distribution curve~~

~~(Zhang et al., 2018; Lu et al., 2020a), as well as the related health impact (Tian et al., 2022).~~

~~Second, TGEOS is suitable for concentration prediction in more comprehensive scenarios that include multiple precursor emissions from multiple sectors. Specifically, in contrast to RSMs with scarce emission variables, sectoral emissions for 26 precursor emissions encompassing over 18 VOC species are incorporated into this model, which enhances the model's capacity to address more flexible demands of policymakers towards interested emission scenarios. Third, given the significant influence of regional transport on local pollutant concentrations (Qiao et al., 2021) and the inability of current technologies to simultaneously consider the impact of regional transport and detailed emission variables, the effects of adjacent grids consist of emission, meteorological conditions, as well as geo-spatial data are taken into account to ensure the accuracy of predictions. In addition, with the use of the Transformer framework, TGEOS demonstrates significantly enhanced predictive accuracy compared to established machine learning models like Multi-Layer Perceptron (MLP) and Random Forest (RF).~~

In this study, we proposed an efficient emulator of GEOS-Chem v14.2.2 based on Transformer architecture, with the capability to provide "GC-aligned" air quality predictions under future emission scenarios in China. It is referred to as "TGEOS" throughout this paper. Superior to earlier studies, TGEOS is capable to provide informative predictions about critical statistical indicators of monthly PM$_{2.5}$ and O$_3$ concentrations (e.g., 75-percentile and max values), and then have a general understanding of probability distribution of future air pollutants. Compared to solely average estimated by previous methods (Liu et al., 2022; Liu et al., 2023), probability distributions can provide informative frequency distributions of pollutants (Yang and Wu, 2022). Many studies have used probability distribution curves to represent future states of PM$_{2.5}$ (Li et al., 2024) and O$_3$ (Zeng et al., 2022) concentrations in diverse emission scenarios, and to explore any extreme pollution events that are typically represented by the high-end tail of the probability distribution curve (Zhang et al., 2018; Zhang et al., 2020), as well as the related health impact (Tian et al., 2022).

Second, TGEOS is suitable for concentration prediction in more comprehensive scenarios that include multiple precursor emissions from multiple sectors. Specifically, in contrast to previous emulators limited by scarce emission variables (Xing et al., 2011, 2020), sectoral emissions for 26 precursor emissions encompassing over 18 VOC species are incorporated into this model, which enhances the model's capacity to address more flexible demands of policymakers towards interested emission scenarios. Third, given the significant influence of regional transport on local pollutant concentrations (Qiao et al., 2021) and the inability of current technologies to simultaneously consider the impact of regional transport and detailed emission variables, the effects of adjacent grids consist of emission, meteorological conditions, as well as geo-spatial data are taken into account to ensure the accuracy of predictions. In addition, with the use of the Transformer framework, TGEOS demonstrates significantly enhanced predictive accuracy compared to other machine learning models.

RC: ***L215: The authors mention that the number of features is 1045 (Table 2), yet the number of input channels is reported as 1048. It would be helpful to clarify how this inconsistency is handled in practice.***

AR: We appreciate the reviewer's careful reading and have corrected this mistake accordingly. The number "1048" was a typographical error and the correct number of input feature dimension is 1045, as stated in Table 2. We have corrected this inconsistency in the revised manuscript.

2.2.1 Model architecture (L215)

In order to align with the shape of the dataset, the model was designed with ~~1048 input channels and 12 output channels.~~an input feature dimension of 1045 and an output dimension of 12.

RC: ***L230: Figure 1 should be improved to provide more detailed information (e.g., Transformer Module).***

AR: Thank you for your helpful suggestion. We have added a new figure to provide more detailed information about the model architecture (Figure 3). The added figure now includes a clear breakdown of key components such as the input embedding, multi-head self-attention, feedforward network, and output layers. We believe this revision improves the clarity and completeness of the model description.

RC: ***Dataset and methodology: The manuscript does not provide basic information such as hyperparameter tuning procedures and sample size.***

AR: Thank you for pointing this out. We have supplemented the aforementioned deficiencies, as outlined below. They will be included in the supplementary.

Table 4: Hyperparameters tuning for TGEOS.

Name	Tuning range	Best value
Batch size	128, 256, 512, 1024	512
Learning rate	1×10^{-5}, 1×10^{-4}, 1×10^{-3}	1×10^{-4}
Epoch	30, 50, 100, 200	100
Number of attention heads	4, 6, 8	8
Number of Encoder layers	2, 4, 6	6
Hidden dim	128, 256, 512	512

Table 5: Other hyperparameter options for TGEOS.

Hyperparameter	Option
Optimizer	Adam
Loss function	MSE
Sample size for training	1,108,440
Sample size for test	221,688

RC: ***Results and discussions: The model evaluation primarily relies on R2 and MAE. It is recommended to include metrics that assess the model's ability to predict extreme values, such as precision and recall for exceedance events.***

AR: Thank you for this valuable suggestion. We will update the figures with extra metrics, namely MBE and

Table 6: Hyperparameters tuning for RF model.

Name	Tuning range	Best value
N estimators	50, 100, 300, 500	300
Max depth	range(10, 50, 5)	25
Min samples split	range(2, 10, 1)	4
Min samples leaf	range(1, 10, 1)	2

Table 7: Hyperparameters tuning for MLP model.

Name	Tuning range	Best value
Batch size	128, 256, 512, 1024	1024
Learning rate	1×10^{-5}, 1×10^{-4}, 1×10^{-3}	1×10^{-4}
Epoch	30, 50, 80, 100	100
Hidden dims	[1024, 512], [2048, 1024, 512], [2048, 1024, 512, 256]	[2048, 1024, 512, 256]
Dropout rate	range(0.1, 0.6, 0.1)	0.2
Activation function	ReLU, Tanh, GELU	ReLU

precision and recall for predictions exceeding the 90th percentile. We have revised corresponding part of the manuscript. Details are shown in blew:

~~As illustrated in Fig. 5(a), 6(a), and S10(a), there exists a robust statistical correlation between the PM$_{2.5}$ indicators predicted by TGEOS and the actual GC simulations across varying emission scenarios, with R^2 values ranging from 0.976 to 0.995. These results substantiate that PM$_{2.5}$ accurately captures the principal trends and patterns of PM$_{2.5}$ as simulated by GC. Moreover, the evaluation of model prediction errors, as quantified by the Root Mean Square Error (RMSE) and Mean Absolute Error (MAE), reveals relatively low error levels, with RMSE values ranging from 0.985 to 2.110 and MAE values between 0.685 and 3.243. This underscores the reliability of TGEOS predictions in relation to those achieved by the GC simulations. In other words, the predictive capabilities of TGEOS are characterized by a high degree of accuracy and reliability.~~ As illustrated in Fig. 5(a), 6(a), and S10(a), there exists a robust statistical correlation between the PM$_{2.5}$ indicators predicted by TGEOS and the actual GC simulations across varying emission scenarios, with R^2 values ranging from 0.976 to 0.995. These results substantiate that PM$_{2.5}$ accurately captures the principal trends and patterns of PM$_{2.5}$ as simulated by GC. The evaluation of model prediction errors, as quantified by the RMSE and MAE, reveals relatively low error levels, with RMSE values ranging from 0.985 to 2.110 and MAE values between 0.685 and 3.243, demonstrating the predictive capabilities of TGEOS with a high degree of accuracy and reliability. The MBE values are ranging from -1.453 to 1.420 for PM$_{2.5}$, -0.033 to 1.125 for O$_3$, indicating a slight overall deviations in concentration predictions

Table 8: Hyperparameters tuning for CNN model.

Name	Tuning range	Best value
Batch size	128, 256, 512, 1024	256
Learning rate	$1 \times 10^{-4}, 1 \times 10^{-3}, 1 \times 10^{-2}$	1×10^{-2}
Epoch	30, 50, 80	80
Kernel size	2, 3, 5	3
Padding	0, 1, 2	1
Number of channels 2	32, 64, 128	128
Size of FC 1	128, 256, 512	256

compared to corresponding GC simulations. Considering that this bias is relatively small compared to the magnitude of the concentrations, the model can be regarded as nearly unbiased. In addition, to evaluate the capability of TGEOS in capturing extreme events, we employed exceedance metrics based on the 90th percentile threshold of the concentration distribution. The results indicate that the model achieves high precision and recall score for both $PM_{2.5}$ and O_3 indicators, with all these values larger than 0.85. These values suggest that the majority of the predicted exceedance events correspond to actual exceedances, while nearly all true exceedance events are successfully detected. The high and balanced values of both metrics demonstrate that TGEOS is capable of accurately identifying extreme high-value occurrences with low false alarm and miss rates. Moreover, this performance highlights the robustness of the model in reproducing the upper tail of the distribution, which is particularly important for applications focusing on extreme pollution events.

In fact, the model's ability to predict extreme values has been presented by estimating key indicators, such as the 75th percentile, and maximum values of daily concentrations within a month. Our results show that the model performs well in predicting these indicators, with R2 exceeding 0.9 in most scenarios. However, the model's predictions for the maximum values are relatively less accurate due to the inherently high uncertainty associated with extremes. Moreover, it is important to note that the extreme values predicted by TGEOS may still reflect the biases inherited from the GEOS-Chem simulation itself (Heald et al., 2012; Travis and Jacob, 2019), and thus may not fully represent the true observed extremes. To address this limitation, we have initiated a follow-up study in which we developed a bias correction module that can be integrated into TGEOS to improve the prediction accuracy. We believe this direction holds great potential and will be a focus of our future work, and we sincerely hope you will stay tuned for our upcoming studies.

References

Brean, J., Rowell, A., Beddows, D. C., Shi, Z., and Harrison, R. M.: Estimates of future new particle formation under different emission scenarios in Beijing, Environmental Science & Technology, 57, 4741–4750, 2023.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale, arXiv preprint arXiv:2010.11929, 2020.

Fang, L., Jin, J., Segers, A., Liao, H., Li, K., Xu, B., Han, W., Pang, M., and Lin, H. X.: A gridded air quality forecast through fusing site-available machine learning predictions from RFSML v1. 0 and chemical transport model results from GEOS-Chem v13. 1.0 using the ensemble Kalman filter, Geoscientific Model Development, 16, 4867–4882, 2023.

Heald, C. L., Collett Jr, J., Lee, T., Benedict, K., Schwandner, F., Li, Y., Clarisse, L., Hurtmans, D., Van Damme, M., Clerbaux, C., et al.: Atmospheric ammonia and particulate inorganic nitrogen over the United States, Atmospheric Chemistry and Physics, 12, 10 295–10 312, 2012.

Huang, L., Liu, S., Yang, Z., Xing, J., Zhang, J., Bian, J., Li, S., Sahu, S. K., Wang, S., and Liu, T.-Y.: Exploring deep learning for air pollutant emission estimation, Geoscientific Model Development Discussions, 2021, 1–22, 2021.

Jin, J., Fang, L., Li, B., Liao, H., Wang, Y., Han, W., Li, K., Pang, M., Wu, X., and Lin, H. X.: 4DEnVar-based inversion system for ammonia emission estimation in China through assimilating IASI ammonia retrievals, Environmental Research Letters, 18, 034 005, 2023.

Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S., and Shah, M.: Transformers in vision: A survey, ACM computing surveys (CSUR), 54, 1–41, 2022.

Li, D., Wu, Q., Cheng, H., Feng, J., Li, D., Wang, Y., Cao, K., and Wang, L.: Numerical study of the future PM2. 5 concentration under climate change and Best-Health-Effect (BHE) scenario, Environmental Pollution, p. 124391, 2024.

Li, Y. and Moura, J. M.: Forecaster: A graph transformer for forecasting spatial and time-dependent data, arXiv preprint arXiv:1909.04019, 2019.

Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J.: A survey of convolutional neural networks: analysis, applications, and prospects, IEEE transactions on neural networks and learning systems, 33, 6999–7019, 2021.

Liu, Z., Dong, M., Xue, W., Ni, X., Qi, Z., Shao, J., Guo, Y., Ma, M., Zhang, Q., and Wang, J.: Interaction patterns between climate action and air cleaning in China: a two-way evaluation based on an ensemble learning approach, Environmental Science & Technology, 56, 9291–9301, 2022.

Lu, X., Zhang, L., Wang, X., Gao, M., Li, K., Zhang, Y., Yue, X., and Zhang, Y.: Rapid increases in warm-season surface ozone and resulting health impact in China since 2013, Environmental Science & Technology Letters, 7, 240–247, 2020.

Qiao, X., Yuan, Y., Tang, Y., Ying, Q., Guo, H., Zhang, Y., and Zhang, H.: Revealing the origin of fine particulate matter in the Sichuan Basin from a source-oriented modeling perspective, Atmospheric Environment, 244, 117 896, 2021.

Taylor, K. E.: Taylor diagram primer, Work. Pap, pp. 1–4, 2005.

Tian, F., Qi, J., Qian, Z., Li, H., Wang, L., Wang, C., Geiger, S. D., McMillin, S. E., Yin, P., Lin, H., et al.: Differentiating the effects of air pollution on daily mortality counts and years of life lost in six Chinese megacities, Science of the Total Environment, 827, 154 037, 2022.

Travis, K. R. and Jacob, D. J.: Systematic bias in evaluating chemical transport models with maximum daily 8 h average (MDA8) surface ozone for air quality applications: a case study with GEOS-Chem v9. 02, Geoscientific Model Development, 12, 3641–3648, 2019.

Vaswani, A.: Attention is all you need, Advances in Neural Information Processing Systems, 2017.

Wu, C., Wu, F., Qi, T., and Huang, Y.: Hi-transformer: Hierarchical interactive transformer for efficient and effective long document modeling, arXiv preprint arXiv:2106.01040, 2021.

Xing, J., Wang, S., Jang, C., Zhu, Y., and Hao, J.: Nonlinear response of ozone to precursor emission changes in China: a modeling study using response surface methodology, Atmospheric Chemistry and Physics, 11, 5027–5044, 2011.

Xing, J., Zheng, S., Ding, D., Kelly, J. T., Wang, S., Li, S., Qin, T., Ma, M., Dong, Z., Jang, C., et al.: Deep learning for prediction of the air quality response to emission changes, Environmental science & technology, 54, 8589–8600, 2020.

Yang, S. and Wu, H.: A novel PM2. 5 concentrations probability density prediction model combines the least absolute shrinkage and selection operator with quantile regression, Environmental Science and Pollution Research, 29, 78 265–78 291, 2022.

Zeng, X., Gao, Y., Wang, Y., Ma, M., Zhang, J., and Sheng, L.: Characterizing the distinct modulation of future emissions on summer ozone concentrations between urban and rural areas over China, Science of the Total Environment, 820, 153 324, 2022.

Zerveas, G., Jayaraman, S., Patel, D., Bhamidipaty, A., and Eickhoff, C.: A transformer-based framework for multivariate time series representation learning, in: Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining, pp. 2114–2124, 2021.

Zhang, D., Wang, Q., Song, S., Chen, S., Li, M., Shen, L., Zheng, S., Cai, B., Wang, S., and Zheng, H.: Machine learning approaches reveal highly heterogeneous air quality co-benefits of the energy transition, Iscience, 26, 2023.

Zhang, J., Gao, Y., Luo, K., Leung, L. R., Zhang, Y., Wang, K., and Fan, J.: Impacts of compound extreme weather events on ozone in the present and future, Atmospheric Chemistry and Physics, 18, 9861–9877, 2018.

Zhao, B., Wang, S., Xing, J., Fu, K., Fu, J., Jang, C., Zhu, Y., Dong, X., Gao, Y., Wu, W., et al.: Assessing the nonlinear response of fine particles to precursor emissions: development and application of an extended response surface modeling technique v1. 0, Geoscientific Model Development, 8, 115–128, 2015.

Zhao, Y., Wang, G., Tang, C., Luo, C., Zeng, W., and Zha, Z.-J.: A battle of network structures: An empirical study of cnn, transformer, and mlp, arXiv preprint arXiv:2108.13002, 2021.