

Article A Hybrid Autoformer Network for Air Pollution Forecasting Based on External Factor Optimization

Kai Pan^{1,2}, Jiang Lu³, Jiaren Li^{1,2} and Zhenyi Xu^{2,*}



- ² Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei 230088, China
- ³ Hefei Branch of China Telecom Co., Ltd., Hefei 230031, China

* Correspondence: xuzhenyi@mail.ustc.edu.cn

Abstract: Exposure to air pollution will pose a serious threat to human health. Accurate air pollution forecasting can help people to reduce exposure risks and promote environmental pollution control, and it is also an extremely important part of smart city management. However, the current deeplearning-based models for air pollution forecasting usually focus on prediction accuracy improvement without considering the model interpretability. These models usually fail to explain the complex relationships between prediction targets and external factors (e.g., ozone concentration (O_3) , wind speed, temperature variation, etc.) The relationships between variables in air pollution time series prediction problems are very complex, with intricate relationships between different types of variables, often with nonlinear multivariate dependencies. To address these problems mentioned above, we proposed a hybrid autoformer network with a genetic algorithm optimization to predict air pollution temporal variation as well as establish interpretable relationships between pollutants and external variables. Furthermore, an elite variable voting operator was designed to better filter out more important external factors such as elite variables, so as to perform a more refined search for elite variables. Moreover, we designed an archive storage operator to reduce the effect of neural network model initialization on the search for external variables. Finally, we conducted comprehensive experiments on the Ma'anshan air pollution dataset to verify the proposed model, where the prediction accuracy was improved by 2-8%, and the selection of model influencing factors was more interpretable.

Keywords: air pollution forecasting; external factor optimization; genetic algorithm, autoformer; temporal variation; multivariate dependencies; relationships analysis; model interpretability

1. Introduction

In recent years, with the development of urbanization and industrialization, air quality issues have been on the agenda [1], and air pollution has been placed on an increasingly important position in policy formulation and implementation. Air pollution in cities is mainly caused by industrial emissions and transportation, which can produce pollutants such as NO₂, O₃, and SO₂ [2]. According to the 2021 China Ecological Environment Status Bulletin, only 64.3% of China's 339 cities at the prefecture level and above met the ambient air quality standards in 2021. In the Yangtze River Delta region, the average percentage of days with air quality exceeding standards was 13.3%, with O₃ and PM_{2.5} as the primary pollutants accounting for 55.4% and 30.7% of the total exceedance days, respectively. The synergistic treatment of multiple pollutants has become the focus of air pollution prevention and control in China [3].

Air pollution prediction refers to the extraction of information and characteristics from historical air pollution data to predict the future trend of air pollution [4]. Multivariate time prediction means that, for the predicted time series, there may be very many external factors that affect the predicted target, such as the value of some pollutants related to



Citation: Pan, K.; Lu, J.; Li, J.; Xu, Z. A Hybrid Autoformer Network for Air Pollution Forecasting Based on External Factor Optimization. *Atmosphere* **2023**, *14*, 869. https:// doi.org/10.3390/atmos14050869

Academic Editor: Anil Namdeo

Received: 11 April 2023 Revised: 5 May 2023 Accepted: 11 May 2023 Published: 14 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).



the historical value of the target pollutant, as well as the closely related external factors including temperature, humidity, wind direction etc. [5]. Many cities have established monitoring stations in various locations to detect ozone (O₃), nitric oxide(NO), PM_{2.5}, and other detection data. The main source of air pollution from industrial emissions, human activities, transportation, natural causes (wild fires), and other factors, such as weather factors, wind speed, temperature, and humidity, will also affect the settlement of pollutants and, thus, affect their monitoring values. One contaminant may also be a precursor to another, and Figure 1 shows the correlation relationships of different pollutants. Thus, air pollution is affected by many complex factors, and these factors can interact with each other, thus making the prediction of air pollution a difficult problem. If it were possible to predict the place with high pollution probability one or two days in advance, more efficient actions could be taken to alleviate the potential regional pollution [6].



Figure 1. The daily trend of O_3 with $PM_{2.5}$ and precursor NO_x on a certain day at a monitoring station:(**a**) O_3 -PM_{2.5}; (**b**) O_3 -NO_x.

Traditional air pollution prediction research are based on statistical methods, such as the Autoregressive Model (AR), the Moving Average Model (MA), and the Auto-Regression and Moving Average Model (ARMA), as well as the Autoregressive Integrated Moving Average (ARIMA) [7]. Although these methods can model the time series well, they all need the time series to have large smoothness, which requires high requirements on the dataset. However, in air pollution monitoring, there will often be problems such as missing data due to sensor failure, so pre-processing of the data is usually required in the application [8]. However, most statistical methods focus only on air pollution values in their predictions, without considering the changes in pollutant concentrations themselves by other factors, such as weather factors and the effects of other pollutants on them. With the development of artificial intelligence approaches and big data, many research projects have utilized machine learning and deep learning techniques for air pollution prediction. In traditional machine learning methods, Fan et al. [9] used a heuristic algorithm combined with SVM to predict daily diffuse solar radiation in air-polluted regions. S.Gocheva-Ilieva et al. [10] proposed a novel framework for stacked regression based on machine learning to predict the daily average concentrations of particulate matter (PM₁₀), where four base models were built and evaluated. Johansson, C et al. [11] applied different machine learning (ML) algorithms—including Random Forest (RF), Extreme Gradient Boosting (XGB), and Long Short-Term Memory (LSTM)—to improve deterministic predictions of PM_{10} , NO_x , and O_3 for 1, 2, and 3 days at different locations in Greater Stockholm, Sweden.

In the problem of air pollution prediction, the use of sample features in traditional machine learning methods mainly requires expert knowledge in air pollution, which is usually time-consuming and laborious. In particular, different regions have different environmental conditions and different characteristics of air pollution changes, and different structural characteristics of atmospheric flow due to topography and population density [12] make it more difficult to extract relevant features. Moreover, air pollutants have very complex chemical reactions; for example, NO_x are important precursor pollutants in O_3 formation and produce complex photochemical reactions, which makes it difficult to construct complex nonlinear feature mappings through traditional machine learning.

Deep learning has made many promising advances in the field of air pollution prediction and analysis. Shikhovtsev A.Yu et al. [13] used a deep neural network based on GMDH to estimate and predict the characteristics of turbulence intensity in the stratosphere. M.Catalano et al. [14] used Autoregressive Integrated Moving Average with Explanatory Variable(ARIMAX) and Artificial Neural Network (ANN) models to compare the result of urban transportation networks for air pollution peak predictions. The results showed that neural networks were predicting peaks at a superior rate to the ARIMAX model. However, the ANN does not reflect the temporal characteristics of air pollution variation well, where the air pollution variation is highly time-dependent and is closely related to the recent air pollution observation, as well as the previous period of variation. Recurrent Neural Networks (RNNs) can use the output of the previous moment as the input of the next moment to achieve feature extraction and the learning of time series. B.T. Ong et al. [15] proposed a Deep Recurrent Neural Network (DRNN) with a novel pre-training method to predict PM_{2.5} in Japan. However, RNNs suffer from gradient disappearance and gradient explosion when dealing with long time sequence problems. M. Krishan et al. [16] used the Long Short-Term Memory (LSTM) approach to predict O₃, PM_{2.5}, NO_{x} , and CO concentrations at a location in the NCT of Delhi. Li et al. [17] proposed a hybrid CNN-LSTM model by combining a Convolutional Neural Network (CNN) with a Long Short-Term Memory (LSTM) neural network for forecasting the next 24 h of PM_{2.5} concentration in Beijing. In recent years, Transformer has made incredible progress in sequential problem processing by using a multi-headed self-attentive mechanism to obtain point-in-time correlations. Chen et al. [18] combined a CNN with Transformer to predict O₃ concentrations and achieved good results in both short- and long-term predictions. Wenfeng Zheng et al. [19–24] used various deep learning methods to achieve better results on both haze time scale and spatio-temporal prediction.

Several contributions have been made in combining genetic algorithms with neural network models to explore hyperparameters; for example, Rana Muhammad Adnan et al. used ALO to optimize the number of hidden layer neurons and the learning rate of an LSTM [25]. The ANFIS-GBO model used two operators to optimize the learning parameters to improve the prediction accuracy of the ANFIS [26]. In addition, the PSOGWO and PSOGSA algorithms have also been used to optimize the control parameters of the ELM model [27]. The SVR-SAMOA model integrates the Simulated Annealing (SA) algorithm with the Mayor Optimization Algorithm (MOA) to determine the optimal hyperparameters for Vector Regression (SVR) [28], and the ANN-EMPA combines mutation and crossover operators with the ANN to produce robust hybrid prediction model [29]. The CNN-INFO is highly efficient in optimizing complex phenomena with unacknowledged search areas [30].

However, all these machine-learning-based methods are hard to interpret and require manual feature engineering based on a priori knowledge, which is prone to prediction errors. For the models built by deep learning training, although the prediction accuracy is high, their trained weights are of little use to us, because they have little physical meaning for real-world problems. For the air pollution prediction problem, the external variables have a very strong correlation with the prediction target, as shown in Figure 1, where $PM_{2.5}$, NO_x , and O_3 are significantly negatively correlated, and, as the PM2.5 concentration increases, the chemical reaction is suppressed, thus reducing the rate of O_3 production. In addition, the non-homogeneous chemical reactions occurring on the surface of the particles due to the increase of $PM_{2.5}$ concentration also affect the concentration of O_3 , while NO_x is a precursor of O_3 and undergoes photochemical reactions to produce ozone. These correlations can help the authorities in predicting atmospheric pollution while helping them to develop effective policies to mitigate pollution, but this information is difficult to obtain in deep learning, and a model is needed that can explore the impact of current external factors on atmospheric pollution while also predicting them.

All these challenges inspire us to rethink the air pollution prediction problems based on deep learning models with model interpretability. Specifically, a Hybrid Autoformer Network with a Genetic Algorithm Model (GA-Autoformer) was proposed to predict air pollution temporal variation, as well as explore the relationship between external variables and target pollution. The main contributions of the proposed method are summarized as follows:

- (1) A Hybrid Autoformer Network with a Genetic Algorithm Model was proposed to predict the air pollution variation, where the genetic algorithm was used to optimize the external variable weighting problem for different variables that have different effects on the target pollution.
- (2) The Elite Variable Operator was proposed to vote at fixed intervals of generations to find out the variables that have a greater impact on the target prediction to be selected as elite variables, which are explored to perform a more refined search.
- (3) The proposed Archive Storage Operator, using genetic algorithms, led to deviations in the final results due to the influence of the initialization of individual models, where the individuals with better value may be less effective due to initialization and vice versa. The archive mechanism was used to store the individuals with good results and to filter them to get the really good ones.
- (4) We conducted comprehensive experiments on the Ma'anshan air pollution dataset to verify the proposed model, where the prediction accuracy was greatly improved, and the selection of model influencing factors was more interpretable.

The rest parts of this paper is arranged as follows. We describe our study area and data set in detail in Section 2. Sections 3 and 4 review the related work and detail our model. The experiments and result analysis are presented in Section 5. Finally, we discuss and conclude the paper in Sections 6 and 7.

2. Study Area and Data Requirement

2.1. Study Area

Ma'anshan is located in East China (Figure 2), east of Anhui Province, in the lower reaches of the Yangtze River, bordering Nanjing, located between 31°46′42″–31°17′26″ north latitude and 118°21′38″–118°52′44″ east longitude. Furthermore, it is an important node city in the South Anhui International Tourism and Culture Demonstration Zone. The overall terrain of the city is relatively flat, slightly higher in the north and lower in the south. It has a north subtropical monsoon climate with four distinct seasons. Ma'anshan Port is one of the top ten ports on the Yangtze River, and Zhengpu Port is the only 10,000-ton deep-water port in the Jiangbei region of Anhui Province, where an Anhui river–sea intermodal transport hub is being built.

As an important industrial city in the Yangtze River Delta region, Ma'anshan straddles the Yangtze River and has a large impact on the surrounding air quality. Our study area covers residential, industrial, and rurial areas of Ma'anshan. The air quality data collected from the above areas can reflect the main air quality conditions of representative areas in Ma'anshan.



Figure 2. The map on the right identifies the location of the urban area of Ma'anshan, and the red dots represent the location of the monitoring stations where the data are concentrated. Location A is located in Ma'anshan Hudong Road Fourth Primary School (Resident area), Location B is located in Poutang National Resort Park (Rurial area), and C is located in Maanshan Economic and Technological Development Zone (industrial area), which correspond to Location A, Location B, and Location C used in the experiment, respectively.

2.2. Data Requirement

We used the air pollutant data from the air pollution quality testing station in Ma'anshan, Anhui Province, and our predicted targets were ozone concentration (O₃), particulate matter with a particle size below 2.5 microns(PM_{2.5}), and the air quality index (AQI). The final model input includes three kinds of information:

- (1) Air pollution gas detection content, including carbon monoxide (CO), nitrogen dioxide (NO₂), and sulfur dioxide (SO₂).
- (2) Air pollution index, including particulate matter with a particle size below 10 microns (PM10) and total suspended particulates (TSPs).
- (3) Environmental factors of the target AQI station, including wind direction, wind speed, precipitation, vapor pressure, humidity, visibility, atmospheric pressure, and temperature.

The specific data set partitioning and description will be detailed in Section 5.1.

3. Preliminary

3.1. Transformer for Time Series Forecasting

Recently, the Transformer model has achieved very excellent results in natural language processing. The self-attention mechanism enables the network to capture long sequences of features while avoiding the circular structure that exists in RNNs and LSTMs by using the parallel design that allows for a significant reduction in prediction time [31].

The Transformer model is a multi-layer model, including a multi-headed self-attentive layer, followed by a feed-forward layer, plus residual connections and layer normalization, as shown in Figure 3.



Figure 3. Architecture of Transformer.

Due to its great potential in sequence modeling, the Transformer soon attracted the attention of researchers. Wu et al. [32] first applied the Transformer to the influenza outbreak prediction problem. Subsequently, researchers carried out optimization of their model, and Li et al. [33] proposed a convolutional self-attention by producing queries and keys with causal convolution and sparse bias to reduce the computational complexity from $O(L^2)$ to O(LlogL). Zhou et al. [34] proposed the ProbSparse Self-Attention mechanism, called Informer, and designed a generative style decoder to produce long-term forecasting. Xu et al. [35] decomposed the time series into trend parts and seasonal parts and used an auto-correlation mechanism.

3.2. Genetic Algorithm

The genetic algorithm was proposed by Professor Holland in 1975 [36]. It is a computational model that simulates Darwin's biological evolutionary process of genetic selection and natural elimination, and, so far, genetic algorithms have been considered as the basis of intelligent optimization algorithms [37]. Genetic algorithms have been widely used in many fields, such as function optimization, path planning, production scheduling, neural-network-structured searches, and other problems [38].

The basic idea of the genetic algorithm is to borrow the law of biological evolution through reproduction–competition to achieve superiority and inferiority, so that the problem is approached to the optimal solution step-by-step. In the process of problem solving, we modeled the data as similar to genes in an organism, and approximate the optimal solution through genetic, selection, crossover, and mutation operations of genes.

The general process of the genetic algorithm is as follows in Figure 4.



Figure 4. Framework of the genetic algorithm.

(1) Selection

Individuals who are suitable as parents are selected from the population according to certain criteria, and the offspring are reproduced by mating. There are various selection methods, such as the fitness proportion method, roulette method, essence preservation method, etc.

(2) Crossover

Crossover is the operation of swapping two chromosomes into groups (recombination). There are various methods of crossover, such as single-point crossover, multi-point crossover, partial mapping crossover (PMX), sequential crossover (OX), heuristic crossover, etc.

(3) Mutation

Mutation is a change in a gene with a certain probability. Mutation has the function of local search, while crossover has the function of global search compared to variation. The crossover and mutation operations help to maintain the diversity of the population and avoid falling into local optima in the early stage of the search.

From the above three basic operations, selection embodies the competitive evolutionary idea of superiority and inferiority, wherein superior individuals are obtained by crossover and mutation operations. Through several iterations, the individuals approach the optimal solution.

In recent years, genetic algorithms have also played a role in air pollution control. The Artificial Neural Network with Back Propagation (BP) with a middle layer and sigmoid activation function and its hybrid with a Genetic Algorithm (BP-GA) were used to predict PM₁₀ levels by M. Asghari [39]. G. Nunnari et al. [40] combined the use of wavelets and genetic algorithms to search for the best wavelet parameters to predict the daily average of SO₂.

3.3. Problem Formulation

In this section, we provide the air pollution prediction problem definition. With a fixed sliding window, the inputs for external factors (other pollutants, meteorological factors, etc.) are $X_T = (X_1, X_2, \dots, X_{T-1}, X_T)$, where $X_i = (X_i^1, X_i^2, \dots, X_i^{L-1}, X_i^L)$ represents a particular external factor belonging to X_T , and the sliding window size is *L*. Furthermore, we set the atmospheric pollutants history value as $y_T = (y_1, y_2, \dots, y_{T-1})$.

$$\tilde{y_T} = f(X, y) \tag{1}$$

Our goal is to build a model that uses external factors X_T and historical pollution values y to obtain predicted air pollution $\tilde{y}_T = f(X, y)$, where f is the nonlinear mapping model to be learned.

4. Methodology

In this section, we will give a detailed description of the Hybrid Autoformer Network with a Genetic Algorithm model (GA-autoformer). A genetic algorithm was used to explore the influence of external factors on the prediction target, determine the degree of influence of each factor on the target, and then put the approximate optimal results into the neural network model. The back-bone neural network model adopted in this paper is Autoformer, which can obtain the best particles and the best prediction accuracy through n iterations. The overall structure of the model is as shown in Figure 5. We will explain the process of the whole structure in detail in Sections 4.1–4.7. The pseudo code for the whole algorithm is shown in Algorithm 1.



Figure 5. Architecture of GA-autoformer. The whole neural architecture consists of two parts. The right side is the neural network architecture, and the autoformer model is used as back-bone network in this paper. The left side is the genetic algorithm, and the lower left side is two operators that we proposed: Elite Voting Operator and Archive Storage Operator.

Algorithm 1 Frame process of the GA-autoformer model **Require:** D: Multivariate time series dataset on air pollution used in this iteration, *t*: Number of iterations, *p*: Population number, k: Interval iteration of executive operator, M: Neural network model **Ensure:** *W_b*: Best weight Randomly generate $W = [W_1, W_2, W_3, \dots, W_{p-1}, W_p]$ $archive_weight = \emptyset$ $archive_fitness = \emptyset$ *elite* = \emptyset for i = 1 to t do $Fitness_W \leftarrow M(D, W)$ $Fitness_a \leftarrow M(D, archive) \cup Fitness_a$ $W = tournament_selection(W,Fitness_W)$ if *i*%*k*==0&&*i*!=0 then *elite* \leftarrow elite_voting(W',Fitness_W) $archive \leftarrow archive_storage(archive_weight, archive_fitness,W',Fitness_a,Fitness_W)$ end if $W' \leftarrow \text{crossover}(Fitness_W, W', elite)$ $W' \leftarrow \text{mutation}(Fitness_W, W', elite)$ $W \leftarrow \text{generate_population}(Fitness_W, W')$ end for $W_b \leftarrow \text{select_best}(archive)$ return W_h

4.1. Generate Random Individuals

Firstly, *n* individuals are randomly generated as the population, and each individual represents a candidate solution, namely, the weight of external factors.

$$W_{i} = [W_{i}^{1}, W_{i}^{2}, W_{i}^{3}, \dots, W_{i}^{L-1}, W_{i}^{L}]$$
⁽²⁾

where *L* is the number of external variables $i \in (1, 2, 3, ..., p)$, and *p* is the number of populations.

4.2. Calculation of Fitness Values

The weight value of each individual is cross multiplied with the multivariate variable value in the current air pollution data set D, and the obtained result is input into the neural network as a new data set D' to obtain the corresponding prediction accuracy, so the prediction accuracy can be used as the fitness value of the current individual.

4.3. Selection

In order not to lose information, we put the best particle into the new population. Then, we used the Tournament Selection Algorithm to randomly select n individuals from the rest of the population and let these n individuals compete. Then we put the best individuals into the new population. We keep iterating until the population of the new population meets the requirements. This new population is the produced offspring and has a higher fitness value than before. Furthermore, n is usually set as 2.

4.4. Elite Variable Voting Operator/Archive Storage Operator

Elite Variable Voting and Archive Storage are performed every *k* generations. The Elite Variable Voting Operator can find elite variables from excellent individuals and search more finely in the subsequent mutation and crossover. The Archive Storage operator can reduce the instability caused by the randomness of neural model initialization by sending the

weight into the neural network model. The specific implementation details are expanded in Sections 4.8 and 4.9.

4.5. Crossover

Although the average fitness value is improved in the process of selection, it cannot produce new individuals. Crossover mimics the method of biological clock hybridization to produce new varieties, transposes some parts of chromosomes, and uses the random pairing method to determine the parents of individuals.

We adopted the following strategy on the crossover of elite variables: elite variables tend to have high weight values, and if they are crossed with some non-elite, it may lead to the loss of information preserved by elite variables, thus leading to population non-convergence, but moderate elite crossover with non-elite variables may also lead to an increase in population diversity, for which we proposed that elite variables be crossed with non-elite variables in the early stage and only with elite variables in the later stage, thus maintaining population convergence. We treated the first 50% of the iteration as the early stage, where elite variables can cross with non-elite variables, and, in the later stage, elite variables could only cross with elite variables. We adopted shuffle crossover as our crossover method [41]. The flow chart of the crossover is shown in Figure 6.



Figure 6. Description of crossover. We separated the elite and non-elite variables for shuffle crossover and then combined them.

4.6. Mutation

Crossover and selection can ensure that excellent genes are left in each evolution, but this may lead to the local optimization of the whole population. When we cross generate a new chromosome, we can randomly select several genes on the chromosome, and randomly modify the value of genes [42].

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$
(3)

$$W_{i}^{k} = W_{i}^{k} + \gamma * W_{i}^{k} * N(0, 1)$$
(4)

Furthermore, we performed a Gaussian variation on W_i^k , wherein the Gaussian distribution random perturbation term $W_i^k * N(0, 1)$ was added to the original state W_i^k . Equation (3) is the general form of the Gaussian distribution, where N(0, 1) is the standard normal distribution, μ is 0, and σ is 1. In order to reflect the difference between elite and non-elite variables, different weighting coefficients γ were used as in Equation 4. When non-elite variables are selected for variation, $\gamma = 1$. Furthermore, when elite variables are selected for variation, $\gamma = 1$. We can find that the elite variables are more important to find the global optimum. It can make the population jump out of the local optimum and improve the convergence speed.

4.7. Iteration

Through *n* iterations of 1–6, the highest fitness value of the individual in the archive in the last generation was calculated, and the individual was taken as the optimal weight as the return value.

4.8. Elite Voting Operator

In the time series prediction of air pollution, certain variables will have more influence on time series forecasting; we call them elite variables. The elite variables vary in different problems. This operator can automatically find the elite variables and optimize them more finely. The pseudo code for the whole algorithm is in Algorithm 2. The following is the step of the Elite Variable Voting Operator.

dist
$$(W_a, W_b) = \sqrt{\sum_{i=1}^{L} (x_a^i - y_b^i)^2}$$
 (5)

Elite variable voting was conducted every k generations.

- (1) Candidates are selected from the top 30% of the population based on population fitness, and these individuals are the best candidates in the population; they represent the evolutionary direction of the population.
- (2) Among the candidates, we want to have some particles that can lead the candidates to the optimization direction more effectively; we call these particles "chairman". We appoint two particles with highest fitness value among the candidates as "chairman". In order to maintain diversity, the candidates that differ most from the "chairman" are added in "chairman". We use the Euclidean distance as Equation (5) to measure the distance between two particles.
- (3) The elite variables chosen by vote are able to give special treatment in the process of mutation and crossover: there is a higher probability of becoming larger in the process of mutation and crossover.

Al	lgorit	hm 2	Elite	Voting	Operator
----	--------	------	-------	--------	----------

Require: W:Population,
$Fitness_W$: Population fitness value
insure: elite: Elite Variables
chairman $\leftarrow \emptyset$
$voters \leftarrow \emptyset$
<i>chairman</i> \leftarrow Take the top 2 fitness values in <i>voters</i> as <i>chairman</i>
<i>chairman</i> \leftarrow <i>chairman</i> +Take the particle in W that is most different from the particle in
chairman voters \leftarrow W- chairman
$elite \leftarrow voting(chairman, voters)$
return <i>elite</i>

4.9. Archive Storage Operator

During the process of neural network training, the result can be bad due to different initialization weights, which can inter the judgment about the effect of weights of different external variables. Therefore, we introduced an archive storage mechanism to put potentially optimal solutions into the archive. The pseudocode for the whole algorithm is in Algorithm 3, and the process is as follows:

Algorithm 3 Archive Storage Operator					
Require: archive_weight,					
archive_fitness,					
W: population,					
$Fitness_W$: Population fitness value,					
<i>Fitness</i> _a : archive fitness value					
Ensure: W: population					
$d \leftarrow \text{size}(archive)$					
$\overline{Fitness_W} = Fitness_W / size(Fitness_W)$					
for $i = 1$ to d do					
if $Fitness_a^i > \overline{Fitness_W}$ then					
$W \leftarrow rejoing_population(W_i, W)$					
else					
$Discard(W_i, archive_weight, archive_fitness)$					
end if					
end for					
return W					

- (1) Every *k* generations, the best particle in all *k* generations is copied into the archive.
- (2) In the next *k* iterations, individuals in archive will not be involved in the process of the genetic algorithm, but only in the calculation of the fitness.
- (3) Every *k* generations, we perform an examination. If the fitness value of the individual is greater than the average fitness value of the current population, this individual will replace the individual with the lowest fitness in the population; otherwise, it will be discarded.

4.10. Prediction and Optimization

For the populations obtained in Sections 4.1–4.7, each particle represents a candidate solution denoted as $W_i = (W_i^1, W_i^2, \dots, W_i^L)$, which represents the weight of each external factor. For the external factor input $X_T = (X_1, X_2, \dots, X_{T-1}, X_T)$, by multiplying each weight with its input counterpart, $X'_t = (W_i^1 * X_t^1, W_i^2 * X_t^2, \dots, W_i^{L-1} * X_t^{L-1}, W_i^L * X_t^L)$. X'_t is fed into the Transformer network along with y.

Unlike traditional forecasting methods that decompose into seasonal parts and trend parts, we gradually decomposed trend and periodic parts from hidden variables in the learning process. This was based on the idea of sliding average, as shown in (6), to achieve progressive decomposition.

$$X'_{t_{trend}} = AvgPool(Padding(X'_{t}))$$

$$X'_{t_{seasonal}} = X'_{t} - X'_{t_{trend}}$$
(6)

After X'_t decomposition into seasonal terms $X'_{t_{seasonal}}$ and trend terms $X'_{t_{trend}}$, the similarity of the different seasonal terms is further aggregated at the encoder pair periodicity using an autocorrelation mechanism. The autocorrelation coefficients can be obtained using the fast Fourier transform, and, finally, the similar subsequence information is aggregated in (7) and (8), where the $\tau_k = \arg \operatorname{Topk}(\mathcal{R}_{\mathcal{Q},\mathcal{K}}(\tau))$. Here, the multi-headed form of query, $\tau \in \{1, \cdots, L\}$

key value, is still used so that the self-attentive mechanism can be replaced seamlessly.

Furthermore, the most probable cycle length is $k = \lfloor c \times \log L \rfloor$ to avoid fusion of irrelevant or even opposite subsequences.

$$[H]\widehat{\mathcal{R}}_{\mathcal{Q},\mathcal{K}}(\tau_1),\cdots,\widehat{\mathcal{R}}_{\mathcal{Q},\mathcal{K}}(\tau_k) = \text{SoftMax}(\mathcal{R}_{\mathcal{Q},\mathcal{K}}(\tau_1),\cdots,\mathcal{R}_{\mathcal{Q},\mathcal{K}}(\tau_k))$$
(7)

AutoCorrelation
$$(\mathcal{Q}, \mathcal{K}, \mathcal{V}) = \sum_{i=1}^{k} \operatorname{Roll}(\mathcal{V}, \tau_k) \widehat{\mathcal{R}}_{\mathcal{Q}, \mathcal{K}}(\tau_k)$$
 (8)

In the decoder, the trend and seasonal terms are predicted separately. For the seasonal term, the feature information obtained by using encoder is aggregated into predicted seasonal values. For the trend term, the information is gradually extracted from the predicted hidden variables using the cumulative method. Finally, the predicted values of the trend and periodicity terms are summed to obtain the predicted values.

By calculating the empirical loss between the trained predicted pollutant value \tilde{y}_T and the real pollutant value y_T , we train the entire model. Our loss function is Root Mean Square Error(RMSE); the loss is not only propagated back from the decoder's outputs across the entire transformer model, it is also involved in the selection of the new generation of the population as the fitness value.

5. Experiments and Results

In this section, we give the parameter settings and experimental results, compare them with some current baselines, and attempt to translate the results into interpretable conclusions. We also try to prove and explain the role of each operator through a series of ablation experiments.

5.1. Dataset Descriptions

The details of the datasets are shown in Table 1. We cut the dataset into a training set and test set by 70% and 30%, respectively, in chronological order. In particular, when we made a prediction for one of the targets, the other two targets were entered into the model as external variables. To measure the importance of external factors, we normalized the dataset.

Data	iset	Location A Location B Location C				
Time_ir	nterval	1 h				
Time	span	1 January 2020–6 October 2020				
Predictio	n target	O ₃ , PM _{2.5} , AQI				
	Relevant pollutants	CO, NO_2, SO_2				
	Air pollution index	PM ₁₀ , TSP				
External factors	Weather factors	wind direction, wind speed, precipitation, vapor pressure, humidity, visibility, atmospheric pres- sure, temperature.				

Table 1. Details of the datasets.

The units of CO (carbon monoxide) is mg/m^3 , the units of O_3 , NO_2 , $PM_{2.5}$ and PM_{10} are $\mu g/m^3$, and TSP is total suspended particulate matter (mg/L). The unit of wind speed is m/s, which is the average wind speed in 10 min. Furthermore, wind direction is measured by an anemometer, which is projected to the $[0, 360^\circ]$ interval. Precipitation (mm) refers to the amount of precipitation per hour, visibility (m) is the 10-min average visibility, humidity (%) is the relative humidity, pressure (hPa) is the atmosphere pressure measured by the monitoring point, and temperature (°C) is measured in degrees Celsius. In addition, we performed statistical analysis of the data in Table 2 to avoid the presence of extreme data.

Variables	Measurement
O3	[2, 300]
PM _{2.5}	[30, 320]
AQI	[30, 320]
СО	[0, 5]
NO ₂	[0, 150]
SO ₂	[5, 210]
PM ₁₀	[0, 330]
TSP	[0, 330]
WindDirection	[0, 360]
WindSpeed	[0, 15]
Precipitation	[0, 82.1]
vapor pressure	[0, 54.9]
humidity	[0, 99]
visibility	[0, 31,565]
atmospheric pressure	[0, 1019.8]
temperature	[-0.8, 37.1]

Table 2. Descriptive statistics of the datasets.

The available time period was from 1 January 2020 to 6 October 2020, and Figure 7 shows the distribution of the amount of data by seasons of the available time period.



Figure 7. Distribution of the AQI, PM_{2.5}, and O₃ data by seasons. (a) AQI-PM_{2.5}. (b) AQI-O₃.

5.2. Parameter Setting

The length of the input sequence in the autoformer was 96, the length of the predicted sequence was 24, the number of headers was 8, the value of dropout was 0.05, the batch_size was 32, and the learning rate was 0.0001. In the genetic algorithm, the size of population was 20, the number of iterations was 50, the probability of crossover was 0.8, the probability of variation was 0.1, and the size of the archive was 5. We set k to be 10.

5.3. Evaluation Metrics

We chose the Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) as the criteria for evaluating the prediction performance.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2}$$
(9)

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |\hat{y}_i - y_i|$$
(10)

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{\hat{y}_i - y_i}{y_i} \right|$$
(11)

where *n* is the length of the time series prediction, y_i is the target value of the model prediction, and y_i is the actual target value.

5.4. Baselines

To verify the performance of our proposed model, we compared GA-autoformer with the following baseline models.

- (1) RNN: RNN is a classical time series prediction model that is capable of extracting time series features. Unlike feed-forward neural networks, which use the output of the previous neuron as input to the next neuron, RNN involves a structure that gives the network the ability to remember information about trends and cycles [15].
- (2) LSTM: LSTM belongs to the class of RNNs and also belongs to the recurrent network model. LSTM solves the problem that RNNs cannot extract long-term time dependence and uses multiple gate mechanisms to alleviate the gradient explosion and gradient disappearance problems that exist in RNNs [16].
- (3) EA-LSTM: EA-LSTM is based on the attention LSTM and uses the genetic-algorithmbased competitive random search (CRS) instead of gradient-based approach to explore the attention layer weights; thus, it better assigns the weights of features within the time window [43].
- (4) Transformer: Recently, the Transformer model has made a big breakthrough in time series prediction. Unlike the RNN and LSTM, Transformer is not a cyclic sequence model. Its prediction efficiency and its ability to predict long-term time series are greatly improved [32].
- (5) Informer: The authors designed an efficient transformer-based long-time prediction model, named Informer, by proposing a ProbSparse self-attentive mechanism, which utilizes self-attentive distillation to highlight dominant attention by halving the cascade layer input with a generative decoder for the one-time prediction of long-time sequence sequences. A new solution to the long-time sequence prediction problem is provided [34].
- (6) Autoformer: The authors used a deep decomposition architecture. The authors designed sequence decomposition units to embed deep models, implement progressively predictive, auto-correlation mechanisms, discard point-wise connected self-attention

mechanisms, and implement series-wise connected autocorrelation mechanisms to break information utilization bottlenecks [35].

5.5. Analysis of Prediction Result

We trained the GA-autoformer for 50 iterations. Table 3 shows the prediction accuracy comparison with different baselines, and the prediction line graphs are put in Figure 8. From Table 3, we can see that our model had a higher prediction accuracy compared to other baseline models, and in comparison with several baseline models, 23 out of 27 comparisons achieved the first, and the remaining four were in the second. For the LSTM and RNN, which all belong to recurrent network structure, they all had a big gap with our model. The EA-LSTM uses the genetic algorithm to optimize the attention layer, combined with the LSTM, but there was still a gap relative to Transformer. Most of the advantages over the Transformer and its other variants were also achieved, which shows that external variables do affect the time series prediction, and this model successfully found the approximate optimal solution of external variables by using a genetic algorithm to jump out of the local optimum. Furthermore, it can be seen from the Figure 9 that the Archive Storage Operator can effectively reduce the impact of the initialization of the neural network on the prediction model.



Figure 8. Actual normalization value and average predicted value for the test experimental run for each monitor station with different target pollutants. The y-axis represents the predicted values and ground truth, and the x-axis is the predicted time-step.

Model					Location A					
Target	O ₃				PM _{2.5}		AQI			
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	
RNN	0.634	1.341	1.469	0.033	0.052	0.423	0.253	0.481	1.151	
LSTM	0.612	1.313	1.419	0.034	0.047	0.346	0.251	0.477	1.072	
EA-LSTM	0.573	1.226	1.297	0.029	0.041	0.335	0.249	0.473	1.081	
transformer	0.514	1.175	1.069	0.030	0.037	0.311	0.241	0.468	1.011	
informer	0.481	1.056	0.816	0.029	0.284	0.285	0.239	0.461	1.036	
autoformer	0.479	0.990	0.778	0.027	0.294	0.275	0.235	0.462	1.021	
GA- autoformer	0.460	0.981	0.761	0.028	0.027	0.242	0.231	0.453	0.861	
					Location B					
Target		O ₃			PM _{2.5}			AQI		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	
RNN	0.439	0.542	1.235	0.227	0.347	2.090	0.045	0.023	0.301	
LSTM	0.422	0.513	1.238	0.156	0.246	1.654	0.041	0.212	0.284	
EA-LSTM	0.342	0.459	1.239	0.107	0.106	0.715	0.034	0.213	0.265	
transformer	0.333	0.454	1.234	0.031	0.044	0.225	0.036	0.189	0.235	
informer	0.329	0.441	1.250	0.038	0.046	0.216	0.031	0.176	0.229	
autoformer	0.305	0.432	1.205	0.028	0.041	0.208	0.034	0.172	0.217	
GA- autoformer	0.315	0.406	1.215	0.026	0.039	0.215	0.029	0.170	0.207	
					Location C					
Target		O ₃			PM _{2.5}			AQI		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE	
RNN	0.081	0.246	0.510	0.294	0.332	1.513	0.169	0.290	0.360	
LSTM	0.061	0.240	0.412	0.262	0.316	1.479	0.091	0.159	0.311	
EA-LSTM	0.057	0.233	0.346	0.251	0.281	1.431	0.041	0.106	0.281	
transformer	0.056	0.231	0.348	0.241	0.279	1.419	0.033	0.063	0.245	
informer	0.054	0.229	0.339	0.236	0.269	1.410	0.036	0.062	0.240	
autoformer	0.052	0.228	0.336	0.231	0.264	1.369	0.034	0.061	0.239	
GA- autoformer	0.051	0.223	0.328	0.209	0.269	1.356	0.031	0.056	0.238	

Table 3. Model prediction performance on air pollution test set.

The RMSE, MAE, and MAPE are the normalization metrics.

Furthermore, we analyzed the effects of γ , archive size, and k on the experiment. Here, we used the dataset Location A, and the prediction target was O₃. From Figure 10, we can see that the best results were obtained when lambda was set to 1.2, archive size was set to 5, and k was set to 10.

And the results of the training process are visualized in Figure 11. It can be clearly seen in Figure 11 that, as the number of iterations increased, the population gradually converged and evolved in the right direction. Regarding the interpretability of the experimental results, we will elaborate in Section 5.6.



Figure 9. Scatter plot of RMSE vs. standard deviation



Figure 10. Sensitivities analysis of γ , archive size, and k.



Figure 11. The training process of genetic algorithm.

5.6. Model Interpretability

Figure 12 shows the external factor optimization weights from the last generation of the population, as well as the elite variables that were taken at the last time in multiple experiments. Furthermore, the color is more red, which indicates that the factor was more important, while the yellow color indicates the less important factor to the target predicted pollution.



Figure 12. We took the weights of the external variables for six random individuals in the final population; the color is more red, which indicates that the variable was more important.

When selecting O_3 as the predicted target pollution, it can be seen that the optimization individuals found all had higher values on NO₂, temperature, etc. The control of O₃ pollution mainly involves the control of its precursors, which are mainly nitrogen oxides and carbon monoxide. Nitrogen oxides react with surrounding atmospheric ozone and subsequently form nitric acid [44]. The increase in temperature represents a flanking increase in solar radiation, which leads to an increase in ozone levels, but high temperatures also lead to increased vertical convective activity in the atmosphere, which facilitates local ozone and precursor diffusion dilution. High humidity facilitates O₃ pollution removal. On the other hand, water vapor in the atmosphere affects solar ultraviolet radiation and, thus, slows down photochemical reactions, and the humidity has a large negative correlation with O₃. Furthermore, O₃ pollution was mainly negatively correlated with wind speed, mainly because wind speed enhances the horizontal diffusion of ozone and contributes to ozone dilution. Those factors' weights were relatively large in the experiments and were all selected as elite variables several times, which is in accordance with our a prior studies on O₃ [45].

In terms of meteorological factors, $PM_{2.5}$ was positively correlated with air temperature and relative humidity and negatively correlated with wind speed. When the wind speed is low and the humidity is high, the intensity of inversion temperature increases, which is unfavorable to the diffusion of $PM_{2.5}$ and other pollutants in the vertical and horizontal directions and aggravates the accumulation of particulate matter pollution, thus making its mass concentration remain high, and when the temperature and relative humidity are both at high levels in autumn and winter, fog is easily produced. The suspended fog droplets easily adsorb and capture gaseous pollutants and particulate matter pollutants, which is favorable to the formation of secondary particles. The hourly concentration of NO_2 had a good positive correlation with the hourly concentration of $PM_{2.5}$, thus indicating that the contribution of traffic pollution emissions to $PM_{2.5}$ was larger. Traffic exhaust emissions are transformed into secondary particles after a period of chemical reaction, which affects the concentration level of $PM_{2.5}$. The results can also be clearly seen in the heat map [46].

When selecting the AQI as the predicted target, it is easy to know that there is a large relationship with SO_2 , O_3 , $PM_{2.5}$, and PM_{10} . As an indicator to measure air pollution, the AQI is closely related to the content of each pollutant. Not only that, wind speed, temperature, and humidity can usually affect the diffusion rate of atmospheric pollutants, and they had strong correlations with atmospheric pollutants, thus leading to strong correlations between the AQI and these factors. It can be seen that various variables were selected as elite variables several times [47].

In addition, the weights derived from the data sets at different locations differed when predicting the same target. In the industrial area (Location C), nitrogen oxide emissions were much larger than in the residential area (Location A) and had a greater weighting compared to the residential area where nitrogen oxide had a greater impact on pollutants. The main sources of pollutants in residential areas are domestic cookers and winter heating, which mostly consume coal and produce carbon monoxide and sulfide, and their corresponding weights were also higher.

From the above analysis, it can be seen that, in the prediction of different targets, our models successfully identified the relationship between external variables and the predicted target pollution, which is consistent with the knowledge of the relevant research, thus proving that the evolutionary direction of the final population of the genetic algorithm is correct. Through the exploration of different external variables, we can analyze and identify the sources of pollutants and help the government develop effective pollution mitigation policies.

5.7. Ablation Experiment

To verify the effects of different genetic algorithm operators, we compared the base transformer with three variant models combined with genetic algorithm optimization, including autoformer (base model), GA-autoformer (our proposed model), autoformer using only the unchanged genetic algorithm (denoted as autoformer-GA(u)), the model using only the Elite Variable Operator(autoformer-GA(elite)) and the Archive Storage Operator (autoformer-GA(archive)), respectively, which are presented in Table 4.

From Table 4, we can find that the Transformer used only the traditional genetic algorithm (autoformer-GA(u)), which had some improvement comparing with the base Transformer. However, the improvement was not significant, and the effects were not as good as our proposed model(autoformer-GA). Furthermore, the two models using only one operator (autoformer-GA(elite) and autoformer-GA(archive)) were not as effective as the proposed model.

Model					Location A				
Target	O ₃		PM _{2.5}			AQI			
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
autoformer	0.479	0.995	0.778	0.027	0.029	0.275	0.235	0.462	1.021
autoformer- GA	0.478	0.991	0.782	0.036	0.029	0.269	0.234	0.461	0.979
autoformer- GA(elite)	0.476	0.986	0.771	0.029	0.029	0.252	0.232	0.457	0.957
autoformer- GA(archive)	0.469	0.984	0.769	0.028	0.028	0.261	0.234	0.451	0.892
our- model	0.460	0.981	0.761	0.028	0.027	0.242	0.231	0.453	0.861

 Table 4. Ablation experiment.

					Location B				
Target	O ₃			PM _{2.5}			AQI		
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
autoformer	0.305	0.432	1.205	0.028	0.041	0.208	0.034	0.172	0.217
autoformer- GA	0.316	0.453	1.233	0.028	0.037	0.218	0.034	0.173	0.208
autoformer- GA(elite)	0.311	0.443	1.234	0.029	0.038	0.195	0.033	0.171	0.205
autoformer- GA(archive)	0.309	0.441	1.234	0.027	0.037	0.204	0.032	0.173	0.202
our- model	0.315	0.406	1.215	0.026	0.039	0.215	0.029	0.170	0.207
					Location C				
Target		O ₃			PM _{2.5}			AQI	
	MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
autoformer	0.052	0.228	0.336	0.231	0.264	1.369	0.034	0.061	0.240
autoformer- GA	0.053	0.226	0.331	0.221	0.261	1.359	0.032	0.058	0.239
autoformer- GA(elite)	0.052	0.229	0.337	0.218	0.262	1.362	0.033	0.059	0.237
autoformer- GA(archive)	0.051	0.224	0.329	0.212	0.261	1.358	0.034	0.059	0.237
our- model	0.051	0.223	0.328	0.209	0.267	1.356	0.031	0.056	0.238

Table 4. Cont.

The RMSE, MAE, and MAPE are the normalization metrics.

For the Elite Variable Operator, as can be seen in the heat map Figure 12, the variables selected as elite variables for many times were larger than the other variables, thus indicating that elite variable operators can find variables that have a greater impact on prediction accuracy and give special treatment to make the search more refined when crossover and mutation occur.

For the Archive Storage Operator, it can be seen in Figure 13 that the model using the Archive Storage Operator not only improved the overall prediction, but also the variance was greatly reduced. This is because the fluctuations caused by the randomized weights of the neural network model could potentially affect the accuracy of the air pollution prediction. Therefore, we stored the potentially good particles and evaluated them several times to filter out the individuals which had better fitness and put them back into the population.



Figure 13. The box plot represents the distribution of the final prediction results of different models. The experimental result is the average fitness value of the population for the last iteration using Location A, with ozone as the prediction target.

6. Discussion

For the air pollution time series prediction problem, Johansson, C et al. [11] used various machine learning methods (e.g., Random Forest (RF), Extreme Gradient Boosting (XGB), and Long Short-Term Memory (LSTM)) for multiple pollutants (PM_{10} , NO_x , and O_3) at multiple locations with multi-temporal predictions. It can be seen that, in the time-series prediction problem, different pollutants and different locations had different environmental conditions and air pollution change characteristics, which require certain a priori knowledge. Prediction models using metaheuristics combined with neural networks are also evolving, such as ANN-EMPA [29], which combines crossover and mutation operators with ANN to enhance the prediction models. However, most of these models generally use metaheuristics to optimize the hyperparameters and the structure of the neural network model, but the results obtained from the optimization are not interpretable and do not explain why the optimized hyperparameters can better improve the pollution prediction model.

To address the above problem, we proposed a Hybrid Autoformer Network with a Genetic Algorithm model to predict air pollution temporal variation, as well as explore the relationship between external variables and target pollution. Unlike the above model, our model combines genetic algorithm with autoformer, wherein autoformer has the ability of long-time series prediction, and the genetic algorithm can explore the influence of external variables on the predicted target, which makes our model interpretable.

From Table 3 and Figure 8, we can find that the prediction accuracy of the GAautoformer was higher than other baseline models. As shown in Figure 9, the standard deviation of the GA-autoformer was also lower than the rest of the models, which indicates that our Archive Storage Operator was able to preserve excellent particles in the iterations. Meanwhile, Figure 12 shows the effects exhibited by different external variables for different prediction targets in different prediction locations, such as industrial areas, residential areas, and suburban areas, which demonstrates the robustness and interpretability of our proposed model.

7. Conclusions

In this study, we proposed a Hybrid Autoformer Network with a Genetic Algorithm model to predict air pollution. A genetic algorithm was used to explore the influence of external variables on pollutants, thus making the model explanatory. In addition, we proposed two operators to better explore external variables: the Elite Variable Voting Operator was used to screen out more important external factors as elite variables and search them more finely; the Archive Storage Operator to store outstanding individuals to alleviate model fluctuations caused by random weights in deep learning. Finally, we conducted a comprehensive experiment on the Ma'anshan air pollution dataset to validate the proposed model. In comparison with the current state-of-the-art models, the prediction accuracy was improved by 2–8%, and the selection of model influencing factors was more interpretable.

However, the air pollution prediction model proposed in this work has some shortcomings, such as that the model cannot handle streaming monitoring data and cannot support online learning. Moreover, the external factors were only explored for a single monitoring site. In future work, we will continue to explore the relationship between external variables in air pollution time series prediction across multiple monitoring sites and continue to explore intelligent optimization algorithms, such as genetic algorithms combined with transfer learning models, and apply them to cross-city air pollution forecasting.

Author Contributions: Conceptualization, Z.X. and K.P.; methodology, Z.X. and K.P.; software, K.P.; validation, J.L. (Jiang Lu) and J.L. (Jiaren Li); formal analysis, K.P. and J.L. (Jiang Lu); investigation, K.P.; resources, J.L. (Jiang Lu) and J.L. (Jiaren Li); data curation, K.P.; writing—original draft preparation, K.P.; writing—review and editing, Z.X.; visualization, K.P. and J.L. (Jiang Lu); supervision, Z.X.; project administration, Z.X.; funding acquisition, Z.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China (62103124, 62033012), as well as the Major Special Science and Technology Project of Anhui, China (202003a07020009).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data availability 864 is not applicable as no datasets were generated or analyzed during the current study.

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Organization, W.H. Air Pollution. Available online: https://www.who.int/health-topics/air-pollution. (accessed on 1 November 2022).
- Steinfeld, J.I. Atmospheric chemistry and physics: From air pollution to climate change. *Environ. Sci. Policy Sustain. Dev.* 1998, 40, 26. [CrossRef]
- 2021 China Ecological Environment Status Bulletin. Available online: https://www.mee.gov.cn/hjzl/sthjzk/zghjzkgb/202205 /P020220608338202870777.pdf (accessed on 1 November 2022).
- Fan, J.; Li, Q.; Hou, J.; Feng, X.; Karimian, H.; Lin, S. A spatiotemporal prediction framework for air pollution based on deep RNN. *ISPRS Ann. Photogramm. Remote. Sens. Spat. Inf. Sci.* 2017, 4, 15. [CrossRef]
- Lengyel, A.; Héberger, K.; Paksy, L.; Bánhidi, O.; Rajkó, R. Prediction of ozone concentration in ambient air using multivariate methods. *Chemosphere* 2004, 57, 889–896. [CrossRef]
- Xu, Z.; Cao, Y.; Kang, Y. Deep spatiotemporal residual early-late fusion network for city region vehicle emission pollution prediction. *Neurocomputing* 2019, 355, 183–199. [CrossRef]
- Badicu, A.; Suciu, G.; Balanescu, M.; Dobrea, M.; Birdici, A.; Orza, O.; Pasat, A. PMs concentration forecasting using ARIMA algorithm. In Proceedings of the 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), Antwerp, Belgium, 25–28 May 2020; pp. 1–5.
- Kumar, U.; Jain, V. ARIMA forecasting of ambient air pollutants (O3, NO, NO2 and CO). Stoch. Environ. Res. Risk Assess. 2010, 24, 751–760. [CrossRef]

- 9. Fan, J.; Wu, L.; Ma, X.; Zhou, H.; Zhang, F. Hybrid support vector machines with heuristic algorithms for prediction of daily diffuse solar radiation in air-polluted regions. *Renew. Energy* 2020, 145, 2034–2045. [CrossRef]
- Gocheva-Ilieva, S.; Ivanov, A.; Stoimenova-Minova, M. Prediction of Daily Mean PM10 Concentrations Using Random Forest, CART Ensemble and Bagging Stacked by MARS. *Sustainability* 2022, 14, 798. [CrossRef]
- 11. Johansson, C.; Zhang, Z.; Engardt, M.; Stafoggia, M.; Ma, X. Improving 3-day deterministic air pollution forecasts using machine learning algorithms. *Atmos. Chem. Phys. Discuss.* **2023**, 1–52. [CrossRef]
- 12. Shikhovtsev, A.Y.; Kovadlo, P.G.; Lezhenin, A.A.; Korobov, O.A.; Kiselev, A.V.; Russkikh, I.V.; Kolobov, D.Y.; Shikhovtsev, M.Y. Influence of Atmospheric Flow Structure on Optical Turbulence Characteristics. *Appl. Sci.* **2023**, *13*, 1282. [CrossRef]
- 13. Shikhovtsev, A.Y.; Kovadlo, P.G.; Kiselev, A.V.; Eselevich, M.V.; Lukin, V.P. Application of Neural Networks to Estimation and Prediction of Seeing at the Large Solar Telescope Site. *Publ. Astron. Soc. Pac.* **2023**, *135*, 014503. [CrossRef]
- 14. Catalano, M.; Galatioto, F.; Bell, M.; Namdeo, A.; Bergantino, A.S. Improving the prediction of air pollution peak episodes generated by urban transport networks. *Environ. Sci. Policy* **2016**, *60*, 69–83. [CrossRef]
- 15. Ong, B.T.; Sugiura, K.; Zettsu, K. Dynamically pre-trained deep recurrent neural networks using environmental monitoring data for predicting PM2.5. *Neural Comput. Appl.* **2016**, 27, 1553–1566. [CrossRef]
- Krishan, M.; Jha, S.; Das, J.; Singh, A.; Goyal, M.K.; Sekar, C. Air quality modelling using long short-term memory (LSTM) over NCT-Delhi, India. *Air Qual. Atmos. Health* 2019, 12, 899–908. [CrossRef]
- Li, T.; Hua, M.; Wu, X. A hybrid CNN-LSTM model for forecasting particulate matter (PM2. 5). *IEEE Access* 2020, *8*, 26933–26940. [CrossRef]
- Chen, Y.; Chen, X.; Xu, A.; Sun, Q.; Peng, X. A hybrid CNN-Transformer model for ozone concentration prediction. *Air Qual. Atmos. Health* 2022, 15, 1533–1546. [CrossRef]
- 19. Yin, L.; Wang, L.; Huang, W.; Liu, S.; Yang, B.; Zheng, W. Spatiotemporal Analysis of Haze in Beijing Based on the Multi-Convolution Model. *Atmosphere* **2021**, *12*, 1408. [CrossRef]
- Tian, J.; Liu, Y.; Zheng, W.; Yin, L. Smog prediction based on the deep belief—BP neural network model (DBN-BP). Urban Clim. 2022, 41, 101078. : 10.1016/j.uclim.2021.101078. [CrossRef]
- 21. Yin, L.; Wang, L.; Huang, W.; Tian, J.; Liu, S.; Yang, B.; Zheng, W. Haze Grading Using the Convolutional Neural Networks. *Atmosphere* **2022**, *13*, 522. [CrossRef]
- 22. Zhang, Z.; Tian, J.; Huang, W.; Yin, L.; Zheng, W.; Liu, S. A Haze Prediction Method Based on One-Dimensional Convolutional Neural Network. *Atmosphere* **2021**, *12*, 1327. [CrossRef]
- 23. Liu, Y.; Tian, J.; Zheng, W.; Yin, L. Spatial and temporal distribution characteristics of haze and pollution particles in China based on spatial statistics. *Urban Climate* **2022**, *41*, 101031. :: 10.1016/j.uclim.2021.101031. [CrossRef]
- 24. Wu, X.; Liu, Z.; Yin, L.; Zheng, W.; Song, L.; Tian, J.; Yang, B.; Liu, S. A Haze Prediction Model in Chengdu Based on LSTM. *Atmosphere* **2021**, *12*, 1479. [CrossRef]
- Yuan, X.; Chen, C.; Lei, X.; Yuan, Y.; Adnan, R.M. Monthly runoff forecasting based on LSTM–ALO model. *Stoch. Environ. Res. Risk Assess.* 2018, 32, 2199–2212. [CrossRef]
- Adnan, R.M.; Mostafa, R.R.; Elbeltagi, A.; Yaseen, Z.M.; Shahid, S.; Kisi, O. Development of new machine learning model for streamflow prediction: Case studies in Pakistan. *Stoch. Environ. Res. Risk Assess.* 2021, 36, 999–1033. [CrossRef]
- Adnan, R.M.; Mostafa, R.R.; Kisi, O.; Yaseen, Z.M.; Shahid, S.; Zounemat-Kermani, M. Improving streamflow prediction using a new hybrid ELM model combined with hybrid particle swarm optimization and grey wolf optimization. *Knowl.-Based Syst.* 2021, 230, 107379. [CrossRef]
- Adnan, R.M.; Kisi, O.; Mostafa, R.R.; Ahmed, A.N.; El-Shafie, A. The potential of a novel support vector machine trained with modified mayfly optimization algorithm for streamflow prediction. *Hydrol. Sci. J.* 2022, 67, 161–174. [CrossRef]
- 29. Ikram, R.M.A.; Ewees, A.A.; Parmar, K.S.; Yaseen, Z.M.; Shahid, S.; Kisi, O. The viability of extended marine predators algorithm-based artificial neural networks for streamflow prediction. *Appl. Soft Comput.* **2022**, *131*, 109739. [CrossRef]
- Adnan, R.M.; Dai, H.L.; Mostafa, R.R.; Islam, A.R.M.T.; Kisi, O.; Elbeltagi, A.; Zounemat-Kermani, M. Application of novel binary optimized machine learning models for monthly streamflow prediction. *Appl. Water Sci.* 2023, 13, 110. [CrossRef]
- 31. Wen, Q.; Zhou, T.; Zhang, C.; Chen, W.; Ma, Z.; Yan, J.; Sun, L. Transformers in Time Series: A Survey. arXiv 2022, arXiv:2202.07125.
- 32. Wu, N.; Green, B.; Ben, X.; O'Banion, S. Deep transformer models for time series forecasting: The influenza prevalence case. *arXiv* **2020**, arXiv:2001.08317.
- Li, S.; Jin, X.; Xuan, Y.; Zhou, X.; Chen, W.; Wang, Y.X.; Yan, X. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. In Proceedings of the Advances in Neural Information Processing Systems 32 (NeurIPS 2019), Processing Systems 2019, NeurIPS 2019, Vancouver, BC, Canada, 8–14 December 2019.
- 34. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. In Proceedings of the AAAI, Vurtual, 2–9 February 2021.
- Xu, J.; Wang, J.; Long, M.; Wu, H. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Adv. Neural Inf. Process. Syst.* 2021, 34, 20.
- 36. Sampson, J.R. Adaptation in natural and artificial systems (John H. Holland). SIAM Rev. 1976, 18, 529. [CrossRef]
- 37. Whitley, D. A genetic algorithm tutorial. Stat. Comput. 1994, 4, 65-85. [CrossRef]
- Kumar, M.; Husain, M.; Upreti, N.; Gupta, D. Genetic algorithm: Review and application. Int. J. Inf. Technol. Knowl. Manag. 2010, 2, 451–454. [CrossRef]

- 39. Asghari, M.; Nematzadeh, H. Predicting air pollution in Tehran: Genetic algorithm and back propagation neural network. *J. Data Min.* **2016**, *4*, 49–54.
- 40. Nunnari, G. Modelling air pollution time-series by using wavelet functions and genetic algorithms. *Soft Comput.* **2004**, *8*, 173–178. [CrossRef]
- Caruana, R.A.; Eshelman, L.J.; Schaffer, J.D. Representation and hidden bias II: Eliminating defining length bias in genetic search via shuffle crossover. In Proceedings of the 11th International Joint Conference on Artificial intelligence, Detroit, MI, USA, 20–25 August 1989; Volume 1, pp. 750–755.
- Higashi, N.; Iba, H. Particle swarm optimization with Gaussian mutation. In Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No. 03EX706) IEEE, Indianapolis, IN, USA, 26 April 2003; pp. 72–79.
- Li, Y.; Zhu, Z.; Kong, D.; Han, H.; Zhao, Y. EA-LSTM: Evolutionary attention-based LSTM for time series prediction. *Knowl.-Based Syst.* 2019, 181, 104785. [CrossRef]
- 44. Obolkin, V.; Molozhnikova, E.; Shikhovtsev, M.; Netsvetaeva, O.; Khodzher, T. Sulfur and Nitrogen Oxides in the Atmosphere of Lake Baikal: Sources, Automatic Monitoring, and Environmental Risks. *Atmosphere* **2021**, *12*, 1348. [CrossRef]
- 45. Liu, H.; Liu, J.; Liu, Y.; Yi, K.; Yang, H.; Xiang, S.; Ma, J.; Tao, S. Spatiotemporal variability and driving factors of ground-level summertime ozone pollution over eastern China. *Atmos. Environ.* **2021**, *265*, 118686. [CrossRef]
- Wang, Y.; Liu, C.; Wang, Q.; Qin, Q.; Ren, H.; Cao, J. Impacts of natural and socioeconomic factors on PM2. 5 from 2014 to 2017. J. Environ. Manag. 2021, 284, 112071. [CrossRef]
- 47. Miao, L.; Liu, C.; Yang, X.; Kwan, M.P.; Zhang, K. Spatiotemporal heterogeneity analysis of air quality in the Yangtze River Delta, China. *Sustain. Cities Soc.* **2022**, *78*, 103603. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.