Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Knowledge-Based Systems 258 (2022) 109996

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Ambient air pollutants concentration prediction during the COVID-19: A method based on transfer learning

Shuixia Chen, Zeshui Xu *, Xinxin Wang, Chenxi Zhang

Business School, Sichuan University, Chengdu 610064, China

ARTICLE INFO

Article history:
Received 23 February 2021
Received in revised form 17 September 2022
Accepted 9 October 2022
Available online 17 October 2022

Keywords:
Ambient air pollutants concentration prediction
Transfer learning
Machine learning
COVID-19

ABSTRACT

Research on the correlation analysis between COVID-19 and air pollution has attracted increasing attention since the COVID-19 pandemic. While many relevant issues have been widely studied, research into ambient air pollutant concentration prediction (APCP) during COVID-19 is still in its infancy. Most of the existing study on APCP is based on machine learning methods, which are not suitable for APCP during COVID-19 due to the different distribution of historical observations before and after the pandemic. Therefore, to fulfill the predictive task based on the historical observations with a different distribution, this paper proposes an improved transfer learning model combined with machine learning for APCP during COVID-19. Specifically, this paper employs the Gaussian mixture method and an optimization algorithm to obtain a new source domain similar to the target domain for further transfer learning. Then, several commonly used machine learning models are trained in the new source domain, and these well-trained models are transferred to the target domain to obtain APCP results. Based on the real-world dataset, the experimental results suggest that, by using the improved machine learning methods based on transfer learning, our method can achieve the prediction with significantly high accuracy. In terms of managerial insights, the effects of influential factors are analyzed according to the relationship between these influential factors and prediction results, while their importance is ranked through their average marginal contribution and partial dependence plots.

1. Introduction

At the beginning of 2020, the novel coronavirus pneumonia called COVID-19 is spreading rapidly in China. And then, it becomes a major global public safety and health event [1]. To effectively control the epidemic, researchers have paid great attention to the transmission and treatment methods of COVID-19 [2]. Specifically, since COVID-19 is a respiratory disease, many studies have focused on the correlation analysis between COVID-19 and air pollution, mainly including the impact of reduced human activity due to COVID-19 on air quality [3] and the relationship between the infection with COVID-19 and air pollution [4]. While many relevant influential issues have been widely studied, less progress, however, has been made in ambient air pollutant concentration prediction (APCP) and management strategy during COVID-19. APCP is of great significance for social environmental governance and personal safety protection, and it has become a focal issue for academics and practitioners [5]. Especially during the pandemic of COVID-19, accurate APCP and effective management strategy can help to guide health management and pollutant emission control, thereby reducing the possible impact of air pollution during COVID-19 [3]. This study is motivated by these two streams of research – *the prediction of air pollutant concentration* and *management strategy* – to contribute to APCP research during COVID-19 from both predictive and managerial perspectives. To achieve this, the study proposes an effective model for APCP during COVID-19 to obtain accurate prediction results and useful management implications.

The APCP refers to adopting advanced information technology to monitor and warn air quality based on a large amount of historical data, to achieve "prevention before disease onset" [6]. The commonly used methods for APCP mainly include deterministic methods [7], statistical methods [8], and machine learning (ML) methods [9]. The deterministic method is a kind of simulation model that considers atmospheric chemical diffusion and transportation process. The necessary simulation process of this method comes at cost of computational complexity and the inaccuracy of prediction due to the lack of real historical data [10]. Statistical methods can well leverage this gap by simulating the relationship between influential factors and prediction targets based on historical observations. However, most statistical methods assume this relationship to be linear [9], which is inconsistent with most practical scenarios. Many ML methods that rely on large numbers of samples relax this linear assumption and show

* Corresponding author.
E-mail addresses: chen_shui_xia@163.com (S. Chen), xuzeshui@263.net (Z. Xu), wangxinxin_cd@163.com (X. Wang), zcx950731@163.com (C. Zhang).

https://doi.org/10.1016/j.knosys.2022.109996
0950-7051/

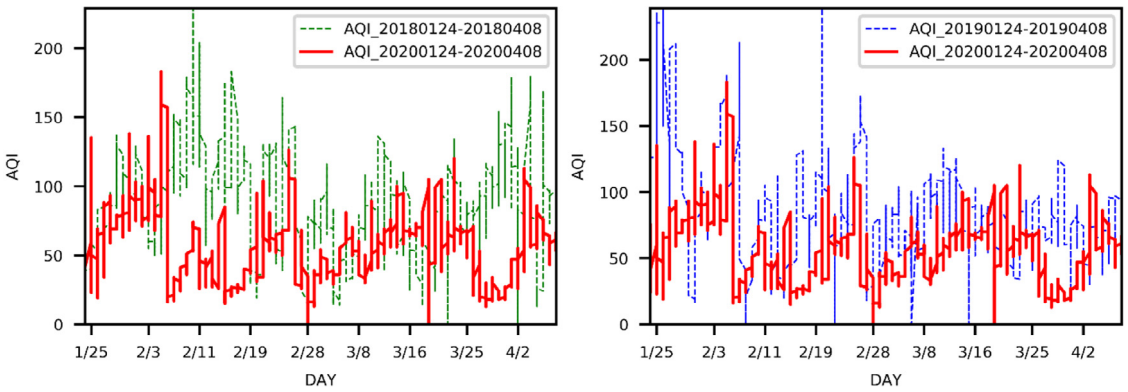

Fig. 1. The AQI of Wuhan during 20180124–20200408.

excellent performance for APCP [11]. This type of method employs the training set divided from the original data set for model training and then applies the well-trained model to the unlabeled test set to obtain the results. Note that almost all ML methods rely on the assumption that the distribution of the training set and test set are similar. Whether ML models could achieve the desired performance depends largely on the amount of sample data to be referred to. That is, only when there are sufficient training sets and test sets with similar distribution can ML methods ensure the accuracy of APCP.

When it comes to APCP during COVID-19, due to some quarantine measures like city lockdowns, there are significant differences in the observed pollutant concentrations before and after the epidemic [12]. Fig. 1 shows the distribution of AQI during 2018–2020 in Wuhan, the first city in China to impose traffic control. Selected January 24, 2020 solstice April 8, 2020, is the time that Wuhan implements the measures of city closure, and corresponding data of 2018 and 2019 are selected for comparison. In Fig. 1, the overall level of AQI in 2020 is lower than that in 2018 and 2019. More importantly, the distribution of AQI in 2020 is different from those in the previous two years. In this case, traditional ML methods are not suitable for APCP during COVID-19: on the one hand, historical observations over the years that could have been used for model training do not meet the requirement of identical distribution; and on the other hand, the current accumulation of identically distributed observations available for the prediction task is limited, particularly in the early stages of the COVID-19 epidemic. Therefore, it is necessary to propose an improved ML model to obtain results of APCP during COVID-19.

Recent researches have started to deal with the limited referred data by adopting transfer learning (TL) to obtain results of APCP [13]. TL is a learning pattern that applies knowledge from related domains (called the source domain, with a large amount of dataset with label) to the applied domain (called the target domain, with a limited dataset even without label), without requiring similar sample distribution between the two domains [14]. For example, Ma et al. [15] proposed a method to transfer the information of the existing air pollutant concentration stations to new stations to obtain the results of APCP in new stations. Ma et al. [9] applied TL to transfer the information of smaller temporal resolutions to larger temporal resolutions to generate high accuracy for APCP at larger temporal resolutions. We find that these existing researches introducing the TL to APCP mainly focus on the spatial transfer or temporal transfer while ignoring the detailed analysis of the relationship between transfer domains. That is, although spatial and temporal transfers are theoretically reasonable, the lack of detailed analysis of transfer domains may result in unconvinced transfers and may affect learning performance. Especially for APCP during COVID-19, the target domain is entirely new and uncharted territory, and it therefore emphasizes the detailed analysis of transfer domains to ensure the feasibility and validity of the prediction.

In this study, we attempt to fill the above research gap by measuring and then minimizing the difference between domains, thereby generating accurate APCP during COVID-19 with the help of TL. Specifically, this study first introduces the Gaussian mixture method (GMM) and maximum mean discrepancy (MMD) to deal with the sample distribution of the source domain and target domain. The former is used to describe the sample distribution of these two domains and the latter calculates the distance between domains represented by GMM. On this basis, this study proposes an optimization algorithm to minimize the difference between domains to obtain a new source domain that is similar to the target domain. Then, this study employs several commonly used ML prediction models to train the new source domain, and these well-trained models are transferred to the target domain to obtain prediction results. Compared to the existing ML prediction models without TL or with other TL strategies, the experimental results suggest that, by using the improved ML methods based on TL, our proposed method can achieve the prediction with significantly high accuracy. In addition to the improvement of prediction, we also complement this study by analyzing some influential factors to obtain management implications. Specifically, this study employs two effective feature analysis methods, i.e., SHapley Additive exPlanations (SHAP) and partial dependence plot (PDP), to explain the relationship between influential factors and results.

The rest of the study is organized as follows: Section 2 reviews the related works. Section 3 proposes the methodology framework. Section 4 develops experimental analysis. Section 5 discusses some result interpretations and managerial insights. Finally, Section 6 concludes this study.

2. Related works

Some basic methods that will be used in the followings are introduced in this section, including TL for knowledge transformation, GMM for domain description, and two feature analysis methods, SHAP and PDP, for result interpretation.

2.1. Transfer learning

Given a labeled source domain $D_s = \{x_i, y_i\}_{i=1}^{n}$ and an unlabeled target domain $D_t = \left\{x_j\right\}_{j=n+1}^{n+m}$ or a labeled target domain with relatively little data $D_t = \left\{x_j, y_j\right\}_{j=n+1}^{n+m}$, data distribution of these two domains, $P(x_s)$ and $P(x_t)$, are different, i.e., $P(x_s) \neq P(x_t)$. The TL is used to find the similarities between these two

domains, thereby achieving the knowledge transferring from the source domain to the target domain to learn the labels [16]. According to the classification of learning methods, the existing TL method can be categorized into instance-based TL, feature-based TL, parameter-based TL, and relation-based TL [17,18].

Instance-based TL focuses on how to select instances from the source domain that are useful for training in the target domain [19]. This type of TL is based on the precondition that there is a similar distribution between source and target domains. Then it can achieve the knowledge transformation by re-weighting samples in the source domain and applying the available information to the target domain. For example, Kim and Lee [20] proposed a new domain adversarial neural network, which can modify the original source domain data and convert it into an auxiliary target domain. By incorporating the attention mechanism into TL, He et al. [21] employed the source domain to create samples for the training of the target domain. Chen et al. [22] adopted the weight updating scheme to obtain valuable samples from the source domain, thereby reducing the effort on the source domain. In our study, the source domain refers to the observations of ambient air pollutant concentration before the COVID-19 epidemic and the target domain refers to the observations during COVID-19. Although there are sample distribution differences between these two domains, they are all time series data related to pollutant emissions. Therefore, the instance-based TL is suitable for the prediction task of this study and we will follow the instance-based transfer strategy to carry out APCP during the COVID-19 epidemic.

Feature-based TL focuses on how to find the common feature representation between the source domain and target domain, and then employs these features for knowledge transformation [23]. The feature-based TL emphasizes effective feature analysis like feature selection, mapping, and encoding. This type of TL method generally includes symmetric and asymmetric feature transformation [24]. The former aims to find valuable features across domains, and the latter reduces the domain difference by transforming the features of the source domain into the target domain. The commonly used transfer component analysis (TCA) is based on symmetric feature transformation, which can discover the representations of cross-domain features by minimizing the differences between marginal distributions [25]. Parameter-based TL assumes that the source domain and target domain share some model parameters or have the same prior distribution. This type of method aims to find these same model parameters or prior distributions to achieve knowledge transformation [26]. For example, the single-model knowledge transfer learns both the knowledge of the target domain and the transfer knowledge in the parameters of the pre-trained model to achieve the prediction task of the target domain [24]. Relation-based TL assumes that if two domains are similar, they will share a similar relationship. Specifically, this method uses the source domain to learn the logical relation network and then applies it to the target domain to achieve knowledge transformation [27]. However, this type of TL method is limited in practice because it involves the complex relational map between source and target domains.

By comparing different types of TL methods, this paper chooses the instance-based TL to obtain results of APCP during COVID-19. Considering that the existing TL methods for APCP have ignored the detailed domain analysis and comparison, this paper introduces a distribution description method, GMM, to obtain the distribution of each domain for further domain analysis and knowledge transformation. A detailed introduction to GMM is shown in the following subsection.

2.2. Gaussian mixture model

The TL method aims to reduce the distribution discrepancy between domains, and it mainly relies on the description of domains with appropriate distributions. For example, the distribution of atmospheric pollutant concentrations is uncertain and may change at any time, and it is therefore impossible to directly measure the difference between sample distributions. GMM is a popular method to explore the distribution structure of samples, which adopts Gaussian distribution to quantify the sample and decompose the sample into several Gaussian sub-models [28]. The domain distribution quantified by GMM can well support the discrepancy calculation between samples, and as a result, it provides a basis for distribution description and knowledge transfer procedure [29]. GMM has been widely used in TL research due to its excellent capabilities for sample distribution analysis [30]. In this study, we utilize this method to obtain the distribution description of historical observation. Specifically, GMM is a simple extension of the Gaussian distribution, which can be regarded as a mixed model composed of K Gaussian sub-distributions (namely, hidden variables). The distribution of GMM can be represented as:

$$F(x) = \sum_{k=1}^{K} \alpha_k \phi(x|\theta_k), \tag{1}$$

where α_k is the probability that the observation data belongs to the kth sub-distribution, $\alpha_k \geq 0$ and $\sum_{k=1}^{K} \alpha_k = 1$. $\phi(x|\theta_k)$ is the Gaussian distribution density function of the kth sub-distribution. And $\theta_k = (\mu_k, \Sigma_k)$, where μ_k, Σ_k represent the mean and covariance matrix of the sample in the kth sub-distribution. K, α_k, μ_k and Σ_k are the parameters needed to be solved. The expectation maximization (EM) algorithm is the commonly used method for parameter training of GMM [31]. This is an iterative algorithm for estimating the maximum likelihood of the parameters of a probability model with hidden variables. Steps for updating parameters of GMM through EM iteration are as follows:

Step 1: Initialize the parameters.

Step 2: Calculate the probability of data j belongs to sub-distribution k, which is represented as r_{jk} and calculated by:

$$r_{jk} = \frac{\alpha_k \phi(x_j|\theta_k)}{\sum_{k=1}^{K} \alpha_k \phi(x_j|\theta_k)}, j = 1, 2, \ldots, N; k = 1, 2, \ldots, K \tag{2}$$

Step 3: Calculate the model parameters for the new iteration.

$$\mu_k = \frac{\sum_j^N (r_{jk} x_j)}{\sum_j^N r_{jk}}, k = 1, 2, \ldots, K, \tag{3}$$

$$\Sigma_k = \frac{\sum_j^N r_{jk} (x_j - \mu_k)(x_j - \mu_k)^T}{\sum_j^N r_{jk}}, k = 1, 2, \ldots, K, \tag{4}$$

$$\alpha_k = \frac{\sum_{j=1}^N r_{jk}}{N}, k = 1, 2, \ldots, K. \tag{5}$$

Step 4: Repeat the above two steps until convergence. Thus, we can obtain the parameters of GMM.

After obtaining the domain distribution by GMM, we further focus on the distribution difference between domains. Commonly used distribution discrepancy measure methods include correlation alignment (CORAL) [32] and MMD [33]. The former realizes deep domain adaptation of knowledge by reducing the difference of covariance matrix between two domains. The latter is a popular unsupervised pattern recognition method, which can calculate the distribution difference between domains by matching appropriate feature representation and kernels [34]. Comparing these two methods, we find that MMD is more compatible with

the domain distributions represented by GMM. In addition, the distance calculation of MMD can support parameter optimization to minimize the difference between domains. Hence, we employ MMD to determine the difference between the two distributions.

2.3. Shapley additive explanations and partial dependence plot

This subsection introduces two types of feature analysis methods to analyze the impact of influential factors on final results, which are helpful to reveal some valuable management implications. First, SHAP is used to rank the Shapley values of features, namely the average marginal contribution of features, to obtain the importance of variables [35], which can be represented by:

$$y_i = y_{base} + g\left(x_{i1}\right) + g\left(x_{i2}\right) + \cdots + g\left(x_{ik}\right), \tag{6}$$

where y_i represents the prediction result and y_{base} represents the baseline for the prediction model (usually the mean of the target variables for all samples). x_i represents the ith sample, and x_{ij} represents the jth feature of the ith sample. $g\left(x_{ij}\right)$ represents the SHAP value of x_{ij}, that is, $g\left(x_{i1}\right)$ is the contribution value of the first feature to the final results y_i in the ith sample. $g\left(x_{i1}\right) > 0$ indicates that this feature shows a positive effect and $g\left(x_{i1}\right) < 0$ represents the negative feature. That is, the results of SHAP generate both feature importance and the polarity of influence, positive or negative [36].

Second, PDP is another type of feature analysis method to further show how features affect the prediction results. That is, different from other feature importance analysis methods, which only focus on the feature importance or the polarity of influence, PDP can show the detailed influence of features on the prediction. This method can help to show how the prediction results vary with the changing of features by capturing the marginal effect of features [37]. Specifically, we can observe the relationship between prediction target and input features, such as whether it is linear, monotonic, or more complex. In addition to obtaining the relationship between the prediction target and input features, the PDP can also be adopted to analyze the relationships between different features.

3. Methodology framework

3.1. Domain distribution comparison

Considering that the distribution of domains greatly influences knowledge transfer, we first explore the distribution comparison of the source domain and target domain. Despite the simple comparison of distribution in Fig. 1, this study employs a useful tool, the Q–Q plot, to show the specific distribution comparison between domains. This plot takes the percentile of each value in the sample data set as the abscissa value, and the percentile of the value in the reference data set as the vertical axis. It employs a 45-degree reference line to visualize distribution differences: if two sample sets come from a population with the same distribution, the sample points should fall near this reference line; on the contrary, the greater the departure from this reference line, the greater the evidence for the conclusion that the two data sets are distributed differently.

This study takes one of the important pollutant indicators, AQI, as an example to show the Q-Q plot in the source domain and target domain, as illustrated in Fig. 2. In this study, the source domain refers to the pollutant observation before the COVID-19 epidemic and the data span is from January 1, 2018, to January 23, 2020. The target domain refers to the pollutant observation during the COVID-19 epidemic and the selected data span is from January 24, 2020, to July 31, 2020. Detailed data introduction is in the following Section 4.1. In Fig. 2, we observe

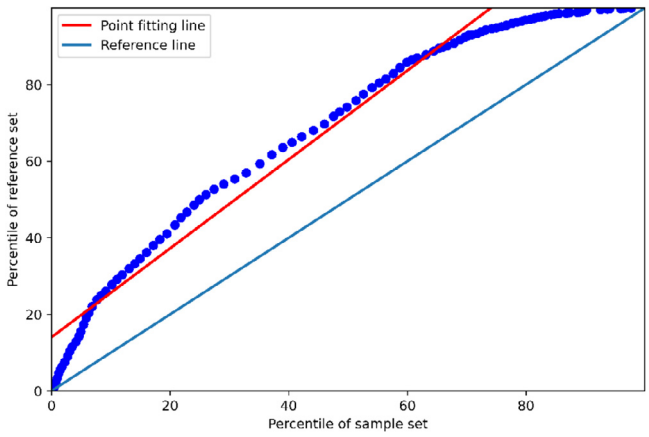

Fig. 2. The Q-Q plot of AQI in the source domain and target domain.

that both the distribution of points and the red line fitted by points deviate significantly from the blue reference line. We also obtain the p-value of the Kolmogorov–Smirnov test, which is a popular method to test whether a group of samples comes from a probability distribution or compare the distribution of the two groups of samples [38]. In this study, the obtained result of the Kolmogorov–Smirnov test is 3.66e−15. In the case of a confidence level of 5%, this result indicates that there is no sufficient reason to explain that the two datasets are subject to the same distribution, which also verifies the results shown in the Q-Q plot. That is, there are distribution differences between these two kinds of domains, and as a result, the traditional ML methods are not suitable for the prediction task. Therefore, this paper introduces an improved TL method combining the analysis of data distribution to achieve the prediction.

3.2. The improved TL prediction model based on GMM

3.2.1. The improved GMM for domain analysis

Facing the significant distribution difference between source and target domains, this study employs the GMM method to determine distributions of source and target domains. According to Eq. (1), the distribution of the source domain represented by GMM is:

$$P_s\left(x_s\left|\theta_s\right.\right) = \sum_{s_k=1}^{s_K} \alpha_{s_k}\phi_{s_k}\left(x_s\left|\theta_{s_k}\right.\right). \tag{7}$$

Let $S_{s_k}\left(\alpha_{s_k}\right) = \alpha_{s_k}\phi_s\left(x_s\left|\theta_{s_k}\right.\right)$, the Gaussian distribution of the above sub-models can be represented as $P_s = \left\{S_{s_k}\left(\alpha_{s_k}\right)\left|\phi_{s_k} \in S_{s_k}, \alpha_{s_k} \geq 0, s_k = 1, 2, \ldots, \#\phi_{s_k}, \sum_{s_k}^{\#\phi_{s_k}} \alpha_{s_k} = 1\right.\right\}$, and $P_s\left(\alpha_s\right) = \left\{S_{s_1}\left(\alpha_{s_1}\right), S_{s_2}\left(\alpha_{s_2}\right), \ldots, S_{s_k}\left(\alpha_{s_k}\right)\right\}$ where $S_{s_k}\left(\alpha_{s_k}\right)$ is $\phi_{s_k}\left(x_s\left|\theta_{s_k}\right.\right)$ related to the weight α_{s_k}, and $\#\phi_{s_k}$ represents the number of Gaussian distribution terms in $\phi_s\left(\alpha_s\right)$. All the notations below have the same meaning. In the same way, the target domain can be represented as:

$$P_t\left(x_t\left|\theta_t\right.\right) = \sum_{t_k=1}^{t_K} \alpha_{t_k}\phi_{t_k}\left(x_t\left|\theta_{t_k}\right.\right). \tag{8}$$

Then Gaussian distribution of the above sub-models can be represented as $P_t = \left\{S_{t_k}\left(\alpha_{t_k}\right)\left|\phi_{t_k} \in S_{t_k}, \alpha_{t_k} \geq 0, t_k = 1, 2, \ldots, \#\phi_{t_k}, \sum_{t_k}^{\#\phi_{t_k}} \alpha_{t_k} = 1\right.\right\}$, and $P_t = \left\{S_{t_1}\left(\alpha_{t_1}\right), S_{t_2}\left(\alpha_{t_2}\right), \ldots, S_{t_k}\left(\alpha_{t_k}\right)\right\}$.

We further solve the parameters involved in the above GMM. As for the parameters s_K and t_K, i.e., the number of clusters

in GMM, this paper employs the Bayesian information criterion (BIC) to obtain the optimal values. BIC is a parameter selection method based on Bayesian considerations, which can effectively prevent excessive complexity by introducing penalty terms [39]. The smaller the value of BIC, the better the performance. And the remaining parameters of GMM can be obtained by Eqs. (2)–(5). After obtaining the source domain and target domain represented by GMM, this study calculates the similarity between these domains to generate a new source domain $\hat{P}_s$ that is similar to the target domain P_t. This study employs MMD for the calculation, and the result $D\left|P_s - P_t\right|$ is:

$$D\left|P_s - P_t\right| = MMD\left(P_s, P_t\right) = \left\| \frac{1}{n}\sum_{s_k=1}^{n} S_{s_k}\left(\alpha_{s_k}\right) - \frac{1}{m}\sum_{t_k=1}^{m} S_{t_k}\left(\alpha_{t_k}\right) \right\|_H^2, \quad (9)$$

where the subscript H represents that this distance is measured by $S()$ mapping the data to the regenerated Hilbert space. Given the distribution difference between the two domains, a new source domain $\hat{P}_s$ can be obtained by adjusting the value of α_{s_k} through the following optimization solution:

$$opt.Min D\left|P_s - P_t\right| \quad st.\begin{cases} \alpha_{s_k} \in [0, 1] \\ \sum \alpha_{s_k} = 1 \end{cases}. \quad (10)$$

By calculating the optimized weights of the sub-distributions in the source domain, a new source domain $\hat{P}_s$ similar to the target domain can be obtained. After that, this paper selects and pre-trains some base models in $\hat{P}_s$, and then transfers them to the target domain to obtain the results of APCP during COVID-19. The architecture of the improved GMM in the study can be concluded as:

First, given the source domain and target domain of the data set to be analyzed, we employ the GMM to obtain the sample distribution of these two domains. Specifically, this study uses the BIC method to obtain the initial number of clusters, i.e., parameters s_K and t_K, and uses the EM method in Section 2.2 to optimize the remaining parameters of GMM, i.e., α, μ, and Σ.

Second, after obtaining the source domain and target domain represented by GMM, i.e., P_s and P_t, this study employs the optimization method based on MMD in Eqs. (9) and (10) to calculate and minimize the discrepancy between the two domains. Then, we can obtain the optimized parameter $\hat{\alpha}_S$ of sub-models in the source domain to generate the new source domain $\hat{P}_s$ with the highest similarity to the target domain.

3.2.2. The TL prediction framework based on the improved GMM

To verify the robustness of the improved TL method, this study selects different types of ML methods to achieve the prediction. Given the continuous prediction setting in this study, we choose five widely used ML regression prediction methods, including linear regression (LR), Bayesian ridge regression (BR), LASSO regression, elastic net regression (ENR), and gradient boosting regression (GBR). LR is the basic regression algorithm based on linear analysis [40]. BR is a ridge-based and Bayesian-based method, which can impose penalties on the size of coefficients to solve some problems with least squares and also has stronger robustness to deal with uncertain problems [41]. LASSO regression is an extension of ordinary regression, which can effectively avoid overfitting by adding an L1 regularization term [42]. ENR is a regression model based on network structure combining L1 regularization (LASSO) and L2 regularization (BR) [43]. GBR is a regression model based on an integration structure and it achieves learning from its mistakes [44]. This model integrates a bunch of poor learning algorithms to learn, so in theory, the results of GBR will be better than those of any other single model. The selected ML methods include general linear regression and its popular variants, regression based on network structure and model integration, basically covering the types of commonly used ML methods. Given the selected ML regression models, the structure of the proposed methodology is shown in Fig. 3. Detailed introduction of each step is as follows:

Step 1: Obtain the sample distribution of the source domain and target domain by GMM. The BIC method and the EM method are adopted to optimize the parameters of GMM. Thus, we can obtain the distribution of the source domain P_s and target domain P_t represented by GMM.

Step 2: Compare the similarity of two domains and optimize parameters to obtain the new source domain similar to the target domain. The optimization method based on MMD in Eqs. (9) and (10) is adopted to calculate and minimize the discrepancy between the two domains, thereby obtaining the new source domain $\hat{P}_s$ with the highest similarity to the target domain.

Step 3: Train the selected base models in the optimized source domain $\hat{P}_s$. The selected five ML regression prediction models, including LR, BR, LASSO, ENR, and GBR, are used as base models and trained in the new source domain.

Step 4: Apply the well-trained models to the target domain and obtain predictive results. The well-trained models in the optimized domain $\hat{P}_s$ are transferred to the target domain P_t to obtain the final prediction results.

4. Experiments

4.1. Data collection

This paper collects monitor data for two years and seven months (ranging from 2018.01.01 to 2020.07.31) from the monitoring station in Wuhan, China. Websites for the monitor data collection include China National Environmental Monitoring Centre (CNEMC) (http://www.cnemc.cn) and National Climatic Data Center (NCDC). The collected data includes the concentration of main ambient air pollutants and some meteorological variables, which are commonly used variables in existing APCP studies. The collected ambient air pollutants/indicators include AQI, PM2.5, PM10, SO2, NO2, O3, and CO. These pollutants/indicators are the key factors used by monitoring stations to report air quality and are also the main reference information used in many existing studies on air pollutants. All the mentioned indicators in addition to AQI are measured by real-time concentration and 24-h moving mean (variables related to O3 include the real-time concentration, 8-h moving mean, 24-h moving mean, and the 24-h maximum of the 8-h sliding mean). The collected meteorological variables include some factors related to atmospheric pollutant concentrations such as temperature, relative humidity, windspeed, wind direction, air pressure, air pressure trend, and amount of precipitation. To simplify the data without losing important information, we collect data at three-h intervals starting from 2:00 every day. And the missing values can be filled by crawling data from adjacent monitoring sites at the same time. A detailed description of the collected data is shown in Table 1.

In this paper, AQI is taken as an example to carry out the prediction. That is, AQI is the dependent variable to be predicted, and other variables in Table 1 are independent variables. Note that AQI data is a kind of time series, the lagged AQI can also be applied as input variables in the prediction model. Considering that most AQI predictions are 3–10 days, we take the commonly used 7 days as the prediction cycle and select the AQI value at the same time 7 days earlier as the additional feature input. In this way, we obtain 7544 pieces of data from 2018.01.01 to 2020.07.31 and 22 features for the prediction task.

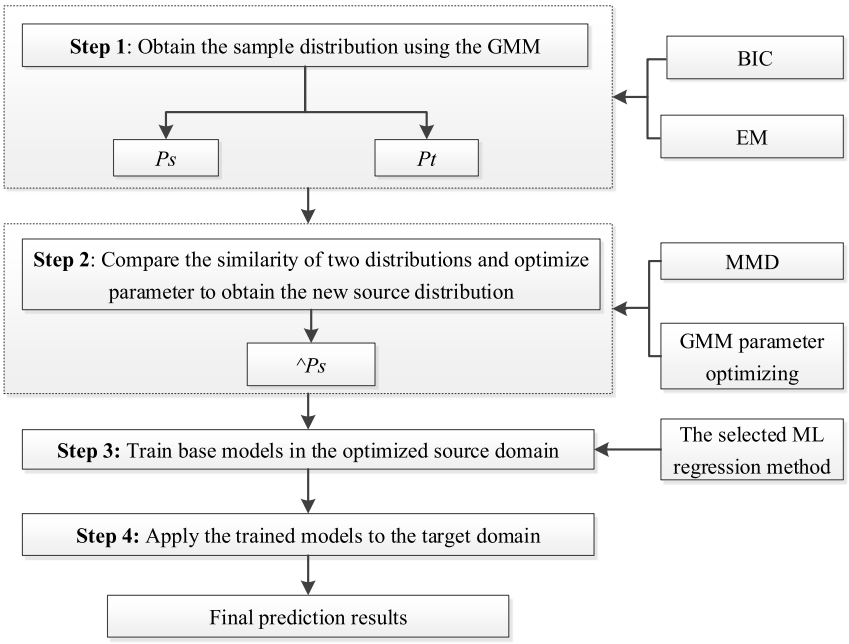

Fig. 3. The structure of the proposed method.

Table 1
A detailed description of the collected data for APCP.

Variable	Description	Major source
AQI	Data that can provide a quantitative description of air quality	/
PM2.5, PM2.5_24h	Real-time concentration and 24 h moving mean of PM2.5	Vehicle emissions, industrial kilns, road and construction dust, straw burning, civil combustion, and so on.
PM10, PM10_24h	Real-time concentration and 24 h moving mean of PM10	
SO2, SO2_24h	Real-time concentration and 24 h moving mean of SO2	The burning of coal, such as coal-fired power plants, and production processes such as non-ferrous metal smelting and sulphuric acid plants.
NO2, NO2_24h	Real-time concentration and 24 h moving mean of NO2	The burning of fossil fuels, including industrial sources such as thermal power generation and motor vehicle exhaust emissions.
O3, O3_8h, O3_24h, O3_8h_24h	Real-time concentration, 8 h moving mean, 24 h moving mean, and the 24 h maximum of the 8 h sliding mean of O3	Motor vehicle exhaust and chemical production.
CO, CO_24h	Real-time concentration and 24 h moving mean of CO	Exhaust gas from motor vehicles, steel making, stoves for civil use, and incineration of solid waste.
Temperature	Real-time observation of temperature	/
Relative humidity	Real-time observation of relative humidity	/
Windspeed	Real-time observation of windspeed	/
Wind direction	Real-time observation of wind direction	/
Air pressure (meteorological station)	Real-time observation of air pressure in the meteorological station	/
Air pressure trend	The change in atmospheric pressure in the three hours before observation	/
Amount of precipitation	Real-time observation of the amount of precipitation	/

4.2. Prediction evaluation method and experiment setting

Given the collected dataset, this paper employs some commonly used methods to evaluate prediction performance, including mean square error (MSE), mean absolute error (MAE), explained variance score (EVS), and R2_score. The calculation of MSE is:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(A_i - F_i)^2\,, \tag{11}$$

where N represents the number of samples. A_i and F_i represent the actual value and prediction value, respectively. The smaller

the value of MSE, the better the fitting effect. MAE is another evaluation method by calculating the average of absolute error between the actual value and prediction value, and the function of MAE is:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|A_i - F_i|. \tag{12}$$

The MAE is commonly used to assess the closeness of the predicted results to the actual value, and the smaller the value, the better the fitting effect. EVS is used to explain the variance scores of regression models, and the value of EVS is [0, 1]. The closer the value of EVS is to 1, the more independent variables can explain variance changes of dependent variables. And the smaller the value, the worse the effect. R2_score refers to the judgment coefficient, which is also the variance score of the regression model. Its value range is [0, 1], and the larger the R2_score, the more independent variables can explain the variance change of dependent variables and vice versa.

We then introduce the experimental settings of this study. All experiments involved in this study are based on Python 3.7. The source codes of the main experiment in this study can refer to https://github.com/chenny1996/APCP-based-on-TL. To satisfy the predictive requirements, the proposed APCP model requires prospective validation, that is, the test set should be isolated from model tuning and forward in time. In this study, we divide the data set in Section 4.1 into a training set containing data from the first 25 months (i.e., source domain, 2018.01.01 to 2020.01.23) and a test set containing data from the last 6 months (i.e., target domain, 2020.01.24 to 2020.07.31). We first employ GMM to deal with the original domains and then obtain the new source domain by the proposed optimized model. Note that all the mentioned analysis of domain distribution in this study only contains the 22 input features without the predicted target AQI. Then this study trains the selected ML prediction models on the new source domain with these input features. Models estimated from the new source domain are then applied to the target domain for AQI prediction during the COVID-19 epidemic. All samples for training are normalized by the minimum–maximum method to ensure the comparable performance of different models.

Like other ML methods, the selected prediction models except LR have hyperparameters that need to be determined during model training, such as the penalty value in LASSO and the maximum depth in GBR. This study determines the optimal hyperparameters for each model by using grid search and 5-fold cross-validation. Then we train the selected models on the target domain and evaluate the prediction performance using the four aforementioned methods: MSE, MAE, EVS, and R2_score. To verify the validity of the proposed model and statistically compare different models, this study employs the *n-out-of-n* bootstrap sample based on the source domain to evaluate each model (including the proposed method and comparison models) 30 times. The *n-out-of-n* bootstrap is an emerging method for deriving test statistics under large samples [45,46], and the comparison results with statistical significance can be obtained by combining this method with paired t-tests. The calculation of paired t-tests of *n-out-of-n* bootstrap is as follows:

Suppose that to compare the predictive performance of model A and model B, N is the number of *n-out-of-n* bootstrap subsets, $P^A = \left[P_1^A, P_2^A, \ldots, P_N^A\right]$ and $P^B = \left[P_1^B, P_2^B, \ldots, P_N^B\right]$ are the obtained prediction performance of model A and model B on all subsets. $P_i = \left(P_i^A - P_i^B\right)$ represents the performance difference between model A and model B on the ith bootstrap subset. The paired t-test calculates the following t statistic for the null hypothesis that the mean difference $\overline{P} = \frac{1}{N}\sum_{i=1}^{N} P_i$ is equal to zero:

$$t = \frac{\overline{P} * \sqrt{N}}{\sqrt{\frac{\sum_{i=1}^{N}(P_i - \overline{P})^2}{N-1}}} \tag{13}$$

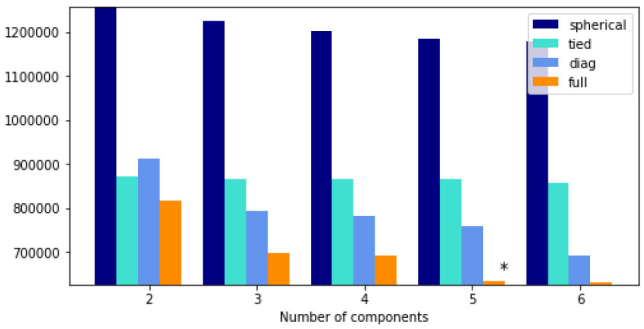

Fig. 4. BIC score of the source domain.

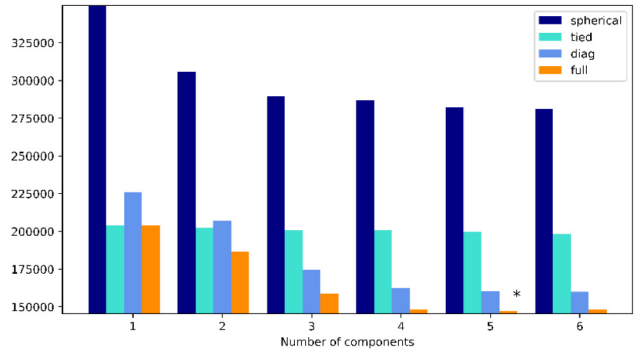

Fig. 5. BIC score of the target domain.

4.3. Results analysis

This study introduces the improved TL prediction model to obtain the results of APCP during COVID-19. We first obtain the sample distribution of the source domain and target domain by GMM. By employing the BIC method, we can obtain the number of clusters in GMM for the source domain and target domain, as shown in Fig. 4 and Fig. 5, respectively. The color-coded boxes in these two figures represent different covariances in GMM. The bar charts marked with * represent the best selection for each domain. We find that the optimal number of clustering of these two domains is 5, and the full covariance performs better.

The remaining parameters of GMM, i.e., parameters of sub-distributions and corresponding weights, can be obtained by EM optimization. Then we can obtain the sample distribution of the source domain and target domain represented by GMM. To visualize the obtained distribution of these samples, we select the first two variables, PM2.5 and PM2.5_24 h, as coordinates to visualize the clustering results of the predicted target AQI. The clustering result of AQI in the source domain is shown in Fig. 6. Due to the partial overlap of the sample distribution, the 3D graph in Fig. 6 still cannot show the results of each cluster. Five clusters of the source domain are separately shown in Figs. A.1 to A.5. The clustering result of AQI in the target domain by GMM is shown in Fig. 7, and separate visualizations are shown in Figs. A.6 to A.10, respectively. To clearly show the data distribution, the mentioned graphs are based on unnormalized data, and the following predictions are based on normalized data to compare different models.

Within the obtained sample distribution of the source domain and target domain, we then compare the similarity of these two domains and obtain the new source domain similar to the target domain by Eqs. (9) and (10). And finally, training selected base models in Section 3.2.2 in the adjusted source domain and applying the well-trained model to the target domain. The results

Table 2
Results of APCP during COVID-19 based on the proposed methods.

Model	MSE	MAE	EVS	R2_score
TL_LR	0.000011 (2.14E−06)	0.002495 (0.000222)	0.965566 (0.006902)	0.958681 (0.010624)
TL_BR	0.000011 (2.14E−06)	0.002495 (0.000222)	0.965561 (0.006901)	0.958684 (0.010625)
TL_LASSO	0.000011 (2.14E−06)	0.002486 (0.000217)	0.965728 (0.006863)	0.958965 (0.010472)
TL_ENR	0.000014 (2.93E−06)	0.002813 (0.000267)	0.954922 (0.008805)	0.947358 (0.012886)
TL_GBR	**0.000008** (4.16E−06)	**0.001819** (0.000226)	**0.971073** (0.009294)	**0.970890** (0.009247)

Note: The numbers in bold represent the best, and the standard deviations of the 30 samples of the Bootstrap test set are in parentheses.

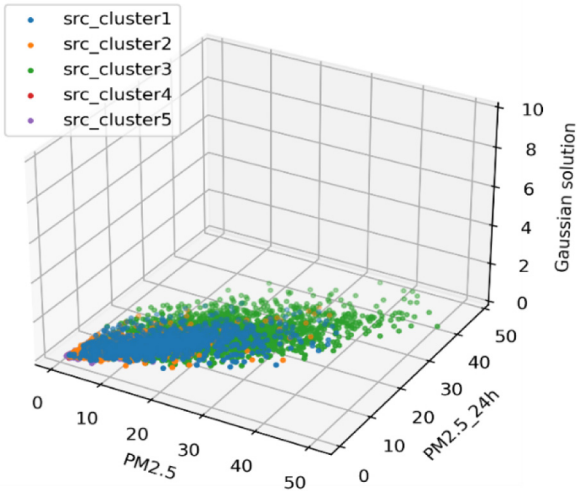

Fig. 6. The clustering result of the source domain obtained by GMM.

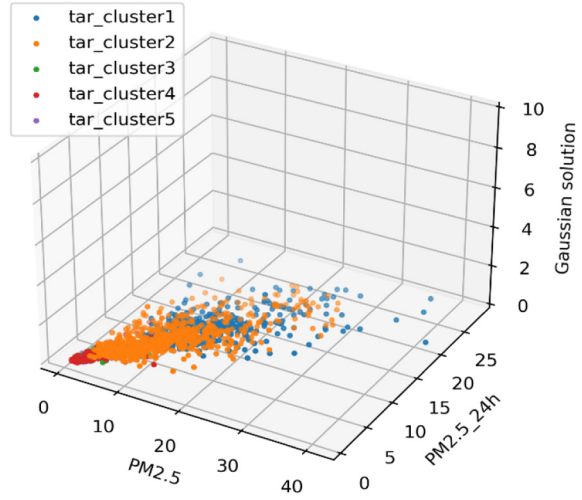

Fig. 7. The clustering result of the target domain obtained by GMM.

obtained by the proposed TL-based ML models are shown in Table 2. The numbers in bold represent the best, and the standard deviations of the 30 samples of the bootstrap test set are in parentheses, the same as below. In these prediction models, we observe that the prediction errors of all models are small and EVS and R2_score are close to 1, which verifies the effectiveness of the proposed model. Further analysis of the table reveals that LR and its variants, BR and LASSO, show little difference in results, while ENR generates the worst results. Among them, the GBR model performs the best, which can be attributed to the effect of model integration in GBR.

4.4. Comparison analysis

4.4.1. Comparison analysis with models without considering TL

To verify the effects of the improved TL model, we compare the prediction results by using corresponding ML methods without TL, as shown in Table 3. Comparing Table 2 with Table 3, we find that the prediction performance of almost all models combined with the improved TL is significantly better than that without TL. The only outlier is that EVS and R2_score of GBR without considering TL are better than that of GBR considering TL. This may be because GBR integrates multiple models, and these two evaluation measures based on variance scores are accordingly biased. From the comprehensive consideration of other models and evaluation measures, we can still conclude that the prediction performance of the proposed model is better than that without considering TL. Further analysis of the comparison finds that although the performance improvement after considering TL is not great in absolute values (this is mainly due to the normalization of the data in our experiments), the relative improvement in prediction performance is considerable. Taking the evaluation method MSE as an example, after combining the optimized TL, the MSEs of the five ML prediction models have decreased 93.125% (= (0.00016 − 0.000011) /0.00016), 93.125% (= (0.00016 − 0.000011) /0.00016), 93.125% (= (0.00016− 0.000011) /0.00016), 91.083% (= (0.000157 − 0.000014) / 0.000157), 70.370% (= (0.000027 − 0.000008) /0.000027), respectively. These results can well verify the effective performance of the proposed model for APCP than models without considering TL.

4.4.2. Comparison analysis with different TL methods

This subsection adopts three other TL methods, TCA, CORAL, and balanced distribution adaptation (BDA) for comparative analysis. TCA is a commonly used marginal distribution adaptation method, the goal of which is to reduce the distance between the marginal probability distribution of the source domain and target domain [25]. CORAL is a statistical feature alignment method, which learns a second-order feature transformation so that the feature distance between the source domain and the target domain can be minimized [47]. BDA is an adaptive method that can leverage the importance of the marginal and conditional distribution differences [48]. While other TL methods can be used for comparison, we believe that the selected TL methods represent different types of knowledge transfer and are all classical methods, making them ideal choices for comparative analysis. With the same training/test data in Section 4.3, we obtain the prediction results of comparison models, as shown in Tables 4 to 8. Each table represents the prediction results of different TL methods with corresponding ML models. Note that due to the long computation time of the TCA model, we did not train this model in the 30 bootstrap samples, and as a result, the standard deviation of this model is not shown in Tables 4–8.

From these tables, we observe that results obtained by the proposed model are better than those of comparison models, and the performance improvement is significant. Note that EVS and R2_score of CORAL_GBR in Table 8 are better than the proposed

Table 3
Results of APCP during COVID-19 based on the selected ML models.

Model	MSE	MAE	EVS	R2_score
LR	0.00016 (8.02E−06)	0.009636 (0.000298)	0.935432 (0.002511)	0.924365 (0.003784)
BR	0.00016 (7.90E−06)	0.009633 (0.000298)	0.935483 (0.002500)	0.924421 (0.003777)
LASSO	0.00016 (8.01E−06)	0.009636 (0.000296)	0.935496 (0.002493)	0.924431 (0.003766)
ENR	0.000157 (7.51E−06)	0.009563 (0.000292)	0.936759 (0.002197)	0.925801 (0.003578)
GBR	**0.000027** (1.75E−06)	**0.004187** (0.000119)	**0.987268** (0.000844)	**0.987134** (0.000804)

Note: The numbers in bold represent the best, and the standard deviations of the 30 samples of the Bootstrap test set are in parentheses.

Table 4
Comparison results of different TL with LR.

Model	MSE	MAE	EVS	R2_score
TL_LR	**0.000011** (2.14E−06)	**0.002495** (0.000222)	**0.965566** (0.006902)	**0.958681** (0.010624)
CORAL_LR	0.000170 (8.8E−06)	0.010005 (0.000322)	0.934656 (0.002567)	0.919831 (0.004135)
BDA_LR	0.000248 (1.26E−05)	0.011916 (0.000321)	0.884492 (0.006344)	0.882716 (0.005937)
TCA_LR	0.001968	0.036357	0.575567	0.069892

Note: The numbers in bold represent the best, and the standard deviations of the 30 samples of the Bootstrap test set are in parentheses.

Table 5
Comparison results of different TL with BR.

Model	MSE	MAE	EVS	R2_score
TL_BR	**0.000011** (2.14E−06)	**0.002495** (0.000222)	**0.965561** (0.006901)	**0.958684** (0.010625)
CORAL_BR	0.000169 (8.74E−06)	0.010002 (0.000322)	0.934707 (0.002557)	0.919886 (0.004129)
BDA_BR	0.000248 (1.15E−05)	0.011914 (0.000282)	0.884634 (0.005841)	0.882762 (0.005431)
TCA_BR	0.001972	0.036427	0.576784	0.067614

Note: The numbers in bold represent the best, and the standard deviations of the 30 samples of the Bootstrap test set are in parentheses.

Table 6
Comparison results of different TL with LASSO.

Model	MSE	MAE	EVS	R2_score
TL_LASSO	**0.000011** (2.141E−06)	**0.002486** (0.000217)	**0.965728** (0.006863)	**0.958965** (0.010472)
CORAL_LASSO	0.000170 (8.798E−06)	0.010006 (0.000322)	0.934654 (0.002569)	0.919826 (0.004137)
BDA_LASSO	0.000246 (1.230E−05)	0.011877 (0.000310)	0.886712 (0.006222)	0.883636 (0.005780)
TCA_LASSO	0.002993	0.046991	0.511742	−0.414699

Note: The numbers in bold represent the best, and the standard deviations of the 30 samples of the Bootstrap test set are in parentheses.

Table 7
Comparison results of different TL with ENR.

Model	MSE	MAE	EVS	R2_score
TL_ENR	**0.000014** (2.927E−06)	**0.002813** (0.000267)	**0.954922** (0.008805)	**0.947358** (0.012886)
CORAL_ENR	0.000167 (8.421E−06)	0.009934 (0.000320)	0.936004 (0.002256)	0.921255 (0.003963)
BDA_ENR	0.000306 (1.884E−05)	0.013723 (0.000497)	0.891421 (0.005186)	0.855462 (0.008920)
TCA_ENR	0.003478	0.050037	0.340819	−0.64406

Note: The numbers in bold represent the best, and the standard deviations of the 30 samples of the Bootstrap test set are in parentheses.

TL_GBR, which can also be explained by the variance score bias of the aggregation model as above. The comprehensive performance of the proposed model still outperforms these comparison models. This result can be explained that some of the comparison models only consider differences in marginal distributions and thus have difficulty in quantifying observed sample changes in APCP. Additionally, some comparison models using other transfer types in addition to instance-based transfer are incompatible with the current learning task. Furthermore, some TL methods like TCA obtain results through matrix calculation, which undoubtedly leads to high computational complexity. The proposed prediction model discusses the differences between domains through data sub-distributions, and the obtained knowledge transfer is more applicable and effective for APCP during COVID-19.

To statistically compare the performance of various models, we evaluate each model in addition to TCA 30 times on the bootstrap samples and the results are shown in Table 9. The values in Table 9 are the t-statistic of the comparison between the model in the row and the corresponding column model, and the asterisk is the significance level of the comparison. As the prediction performance of LR, BR, and LASSO is not significantly different, we only choose LASSO as the representative one in Table 9. In this table, for the benefit evaluation measures, EVS and R2_score, the large positive t-statistic indicates that the model in the row outperforms the corresponding column model, whereas a more negative t-statistic suggests the reverse. For cost evaluation measures, MAE and MSE, the results are opposite.

5. Discussion – Result interpretation and managerial insights

Accurate prediction results are useful for future decision-making, and the detailed analysis of influential factors is also helpful for result interpretation to generate some useful managerial insights. In this section, we employ two methods, SHAP

Table 8
Comparison results of different TL with GBR.

Model	MSE	MAE	EVS	R2_score
TL_GBR	**0.000008** (4.159E−06)	**0.001819** (0.000226)	0.971073 (0.009294)	0.970890 (0.009247)
CORAL_GBR	0.000032 (2.387E−06)	0.00441 (0.000153)	**0.985723** (0.001092)	**0.984691** (0.001106)
BDA_GBR	0.000705 (5.703E−05)	0.021445 (0.000764)	0.796994 (0.016657)	0.666844 (0.026940)
TCA_GBR	0.0013	0.029798	0.719994	0.38538

Note: The numbers in bold represent the best, and the standard deviations of the 30 samples of the Bootstrap test set are in parentheses.

Table 9
Pair comparisons of each model based on bootstrap samples and t-statistic.

		TL_LASSO		TL_ENR		TL_GBR
MSE	CORAL_LASSO	98.1535***	CORAL_ENR	98.31521***	CORAL_GBR	35.20019***
	BDA_LASSO	106.8178***	BDA_ENR	79.52758***	BDA_GBR	67.94562***
MAE	CORAL_LASSO	118.5499***	CORAL_ENR	108.4205***	CORAL_GBR	59.47519***
	BDA_LASSO	140.1118***	BDA_ENR	101.9126**	BDA_GBR	133.2196**
EVS	CORAL_LASSO	−23.7185**	CORAL_ENR	−11.5139**	CORAL_GBR	6.791105**
	BDA_LASSO	−46.3072*	BDA_ENR	−33.6071*	BDA_GBR	−51.7483
R2_score	CORAL_LASSO	−18.9603**	CORAL_ENR	−10.0756**	CORAL_GBR	6.404508**
	BDA_LASSO	−32.977*	BDA_ENR	−34.4672*	BDA_GBR	−60.4539

*$p < 0.1$.
**$p < 0.05$.
***$p < 0.01$.

and PDP, to carry out feature importance analysis. Considering the outstanding performance of TL_GBR in Table 2, this section takes the results of the proposed TL_GBR method to introduce the following analysis.

5.1. Results and discussion of SHAP analysis

Using the open-source package *shap* in Python, we can obtain the results of SHAP for this study. The top 20 important features obtained by SHAP are shown in Fig. 8. In this figure, the samples are represented by points with different colors. The redder the point color represents the greater the feature value, and the bluer the point color represents the smaller the feature value. The ordinate of Fig. 8 represents features and the abscissa is the SHAP value of features, i.e., the influence of features. In this figure, we find that some features show positive effects on prediction, and the representative features include PM10, PM2.5, and O3. And some of the features show no significant effects, that is, the increase or decrease of these types of features will not directly lead to the change of the influence on the results, or the corresponding change has no obvious rule. For example, as for the feature O3_24 h, we observe that when the feature value is small, the SHAP value is close to 0. But when the feature value is large, the changes of SHAP have no obvious rule. And as for some meteorological features, such as wind direction and windspeed, both the increase and decrease of feature value show no significant influence on SHAP.

To obtain a clearer understanding of how each feature affects the final results, Fig. 9 supplements the importance of features obtained by calculating the mean SHAP value. Combining these two figures, we can conclude: (1) Compared with meteorological features, pollutant-related features show more influence on the prediction of AQI; (2) Among pollutant-related features, PM10 and PM2.5 show more significant influence than other pollutants; (3) Some meteorological features that are generally considered important, like windspeed and amount of precipitation, show no significant effects on the prediction. While some meteorological features which are easily ignored, such as air pressure trends, have a great influence on the prediction results. The above results can help us understand the role of each feature in the prediction, and then the PDP is adopted to analyze how these features affect the final results.

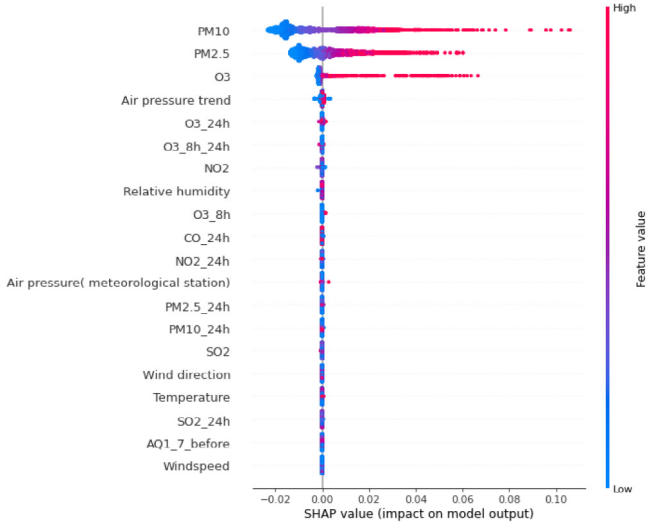

Fig. 8. Results of SHAP value.

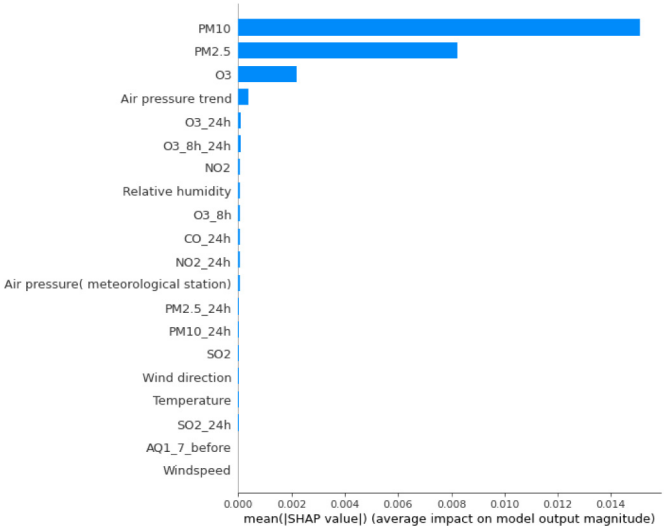

Fig. 9. The importance of features obtained by SHAP.

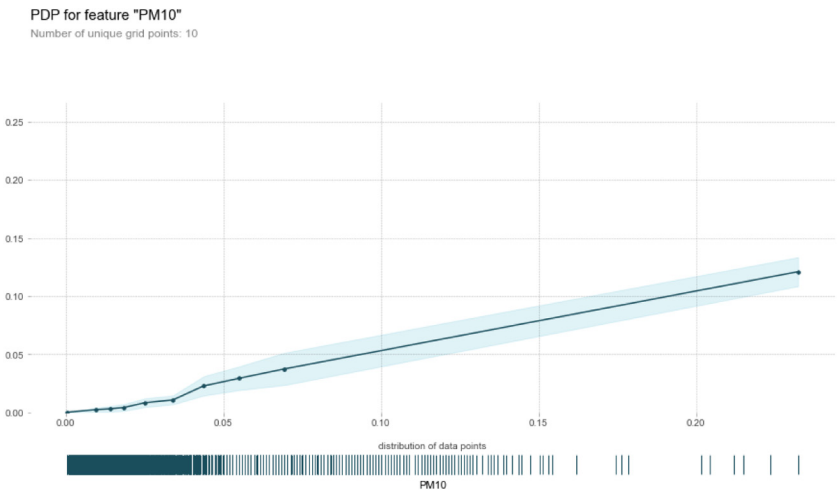

Fig. 10. PDP of PM10.

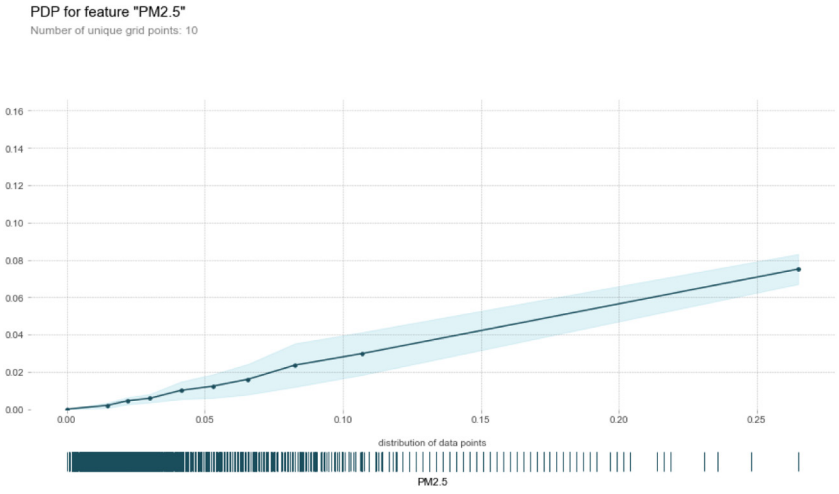

Fig. 11. PDP of PM2.5.

5.2. Results and discussion of PDP analysis

This subsection selects four relatively important features in Fig. 9, PM10, PM2.5, O3, and air pressure trends, to analyze how they affect the prediction. Using the open-source package in Python, *pdp*, we obtain the PDPs of these four features, as shown in Figs. 10 to 13. The *x*-axis of these figures represents the feature and the *y*-axis represents the change of predicted values. The blue shaded part represents the confidence interval. In Figs. 10 to 12, we observe a monotonically increasing linear relationship between the three pollutant-related features and the predicted target. This result indicates that the influence of these features on the predictive target is almost linear, and this influence increases with the increase of the feature value. These findings can provide guidance for managers to make decisions. For example, given the absolute contribution of PM10 and PM2.5 to the prediction, these two variables should be monitored more rigorously to accurately predict AQI values during COVID-19. In addition, the positive linear effects of these variables on prediction remind us to pay more attention to the major source of PM10 and PM2.5 shown in Table 1 and take corresponding control measures to alleviate air pollution. While in Fig. 13, we observe that the positive effect of air pressure trends on prediction is short-lived and disappears when the variable increases to a certain extent. This finding supplements Fig. 9 about how air pressure trends specifically affect the prediction.

The above analysis draws a conclusion about the relationship between each feature and the prediction target. Then we further focus on the analysis of different features to uncover the effect of controlling these features on other features. This study selects the feature with the greatest impact, i.e., PM10, and explores the relationship between PM10 and the other three main features. We employ the PDP to obtain the results, as shown in Figs. 14 to 16. In these figures, the *x*-axis and *y*-axis represent the features to be analyzed, and the *z*-axis represents the influence on the prediction results. In Fig. 14, we observe that when one of PM10 or PM2.5 is fixed, the value of another feature changes will also lead to the corresponding impact on the prediction results. In Fig. 15 and Fig. 16, we observe that when the values of PM10 are constant, the change of O3 and air pressure trends do not cause significant changes in the results. On the contrary, the change in PM10 can lead to significant changes in the results. These observations indicate that both PM2.5 and PM10 have a positive correlation with the predicted results, while O3 and air pressure trends show no significant positive effects on the prediction when fixing PM10. Therefore, we can conclude that controlling PM10 and PM2.5 has a greater impact on the prediction, and more attention should be paid to these two factors.

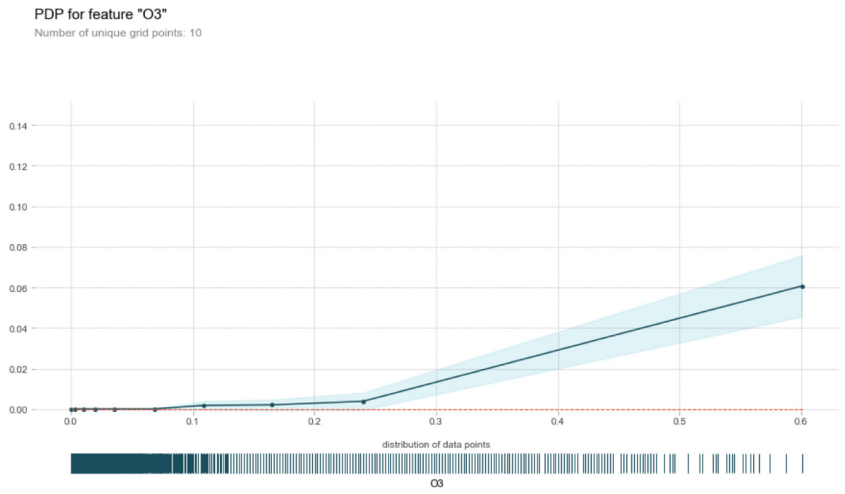

Fig. 12. PDP of O3.

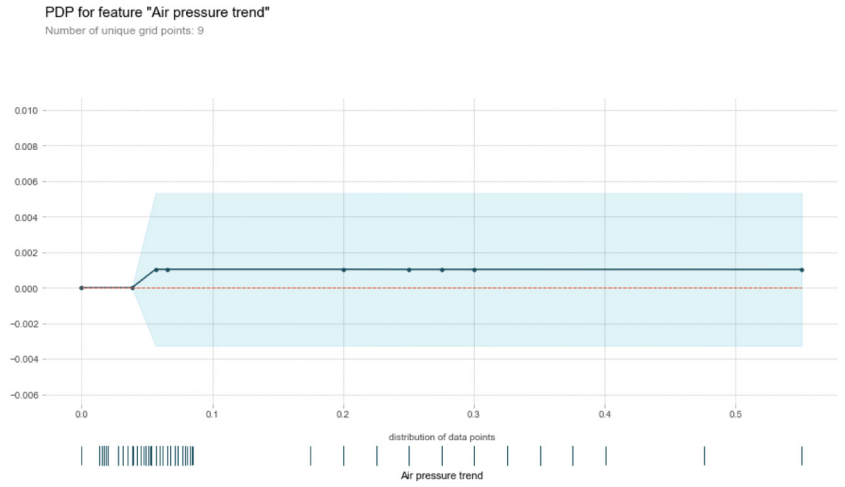

Fig. 13. PDP of air pressure trends.

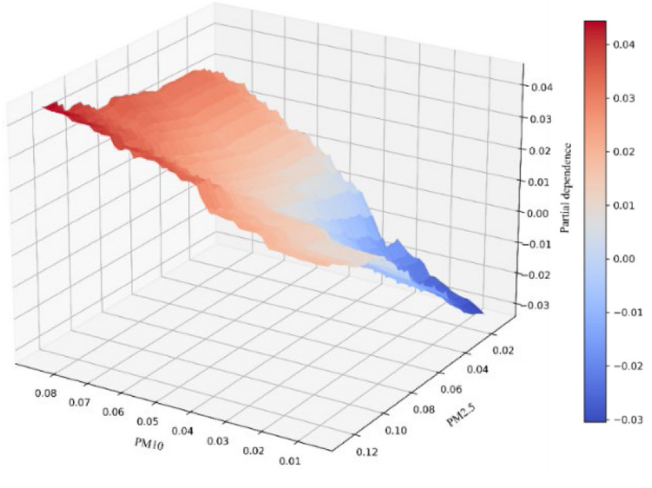

Fig. 14. The PDP relationship analysis between PM10 and PM2.5.

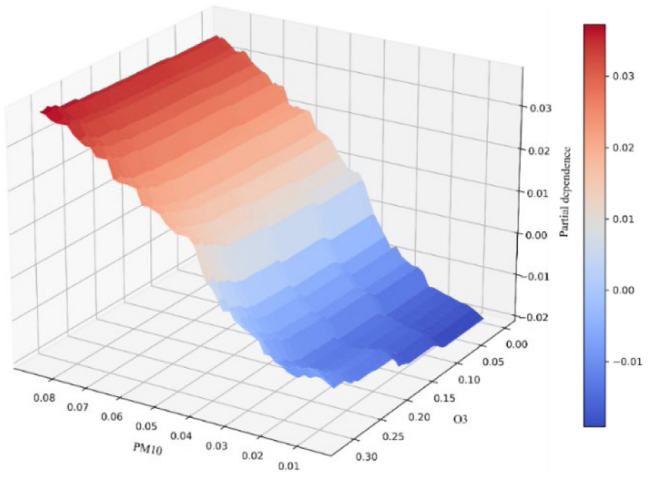

Fig. 15. The PDP relationship analysis between PM10 and O3.

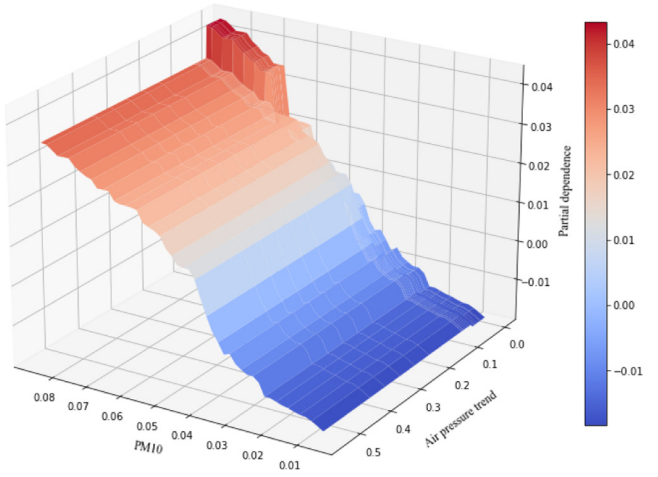

Fig. 16. The PDP relationship analysis between PM10 and air pressure trends.

6. Conclusions

This paper proposes an improved prediction model based on TL for APCP during COVID-19 and analyzes some influential factors to obtain effective management implications. Based on the dataset collected from CNEMC and NCDC, the proposed method shows better predictive performance than other comparison methods. Then, the SHAP and PDP are adopted to explain the relationship between influential factors and prediction results to conclude the following findings:

(1) All features either have a positive effect or show no obvious rule of influence on the prediction results. Specifically, several main features including PM10, PM 2.5, and O3 show significantly positive effects on the prediction of AQI.
(2) Some meteorological features that are generally considered important show no significant effects on the prediction of AQI. Air pressure trends, a feature that is not easily perceived by humans, play a significant role in the prediction. This result reminds us that we should not only focus on features that are considered to be valid subjectively but should fully consider the possible influencing variables.
(3) PM2.5 and PM10 have a positive correlation with the prediction of AQI. Therefore, when taking AQI as the goal of air pollution treatment, managers should pay more attention to the control of PM10 and PM2.5 during COVID-19 in Wuhan.

The model proposed in this paper still has some limitations, which provide directions for future work. First, this paper only considers the TL based on the instance of APCP during COVID-19, and further studies on the TL, such as TL based on deep learning, can be considered in the future. Second, this paper only analyzes the relationship between features and results without considering the impact of actual traffic control measures. Future research should consider the effects of practical restrictions to obtain some practical management insights. Additionally, future research can extend the proposed TL method to multi-source domain transfer and further explore the application of the proposed method in more TL tasks and practical scenarios.

CRediT authorship contribution statement

Shuixia Chen: Data curation, Methodology, Writing – original draft. **Zeshui Xu:** Supervision, Writing – review & editing. **Xinxin Wang:** Writing – review & editing. **Chenxi Zhang:** Visualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors thank the editors and anonymous reviewers for their helpful comments and suggestions. This work was supported by the National Natural Science Foundation of China (No. 72071135).

Appendix

See Figs. A.1–A.10.

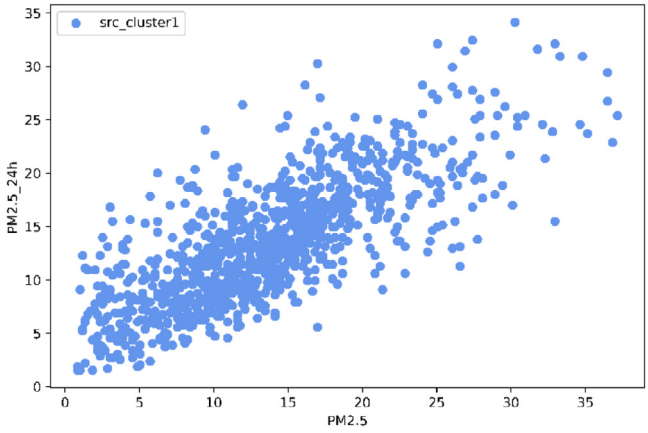

Fig. A.1. The first cluster of source domain obtained by GMM.

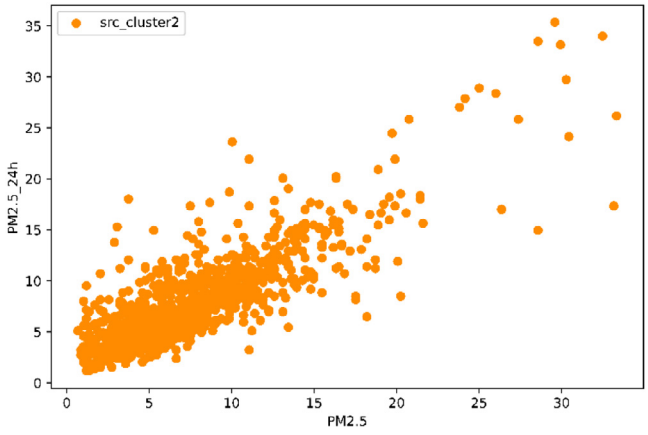

Fig. A.2. The second cluster of source domain obtained by GMM.

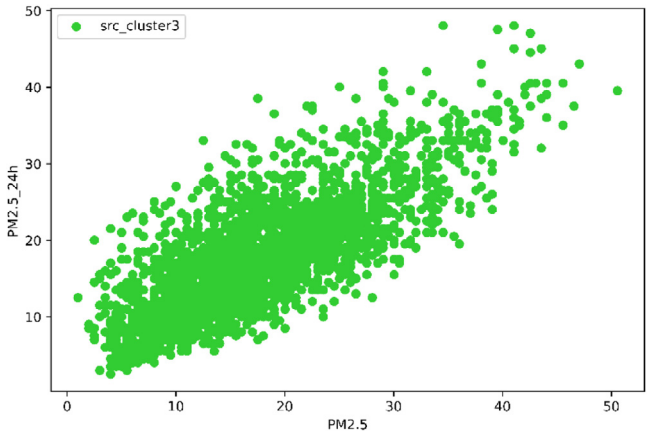

Fig. A.3. The third cluster of source domain obtained by GMM.

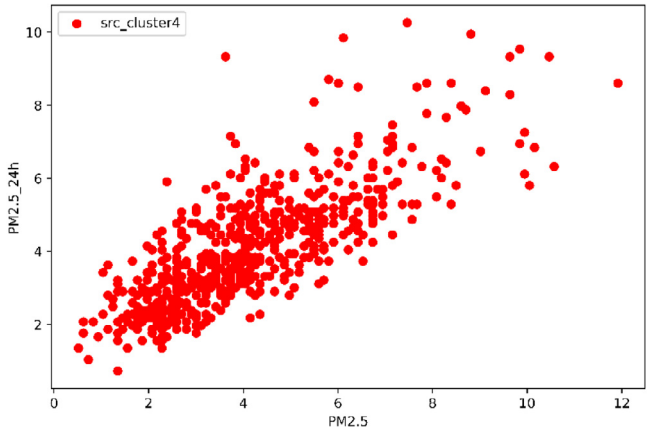

Fig. A.4. The fourth cluster of source domain obtained by GMM.

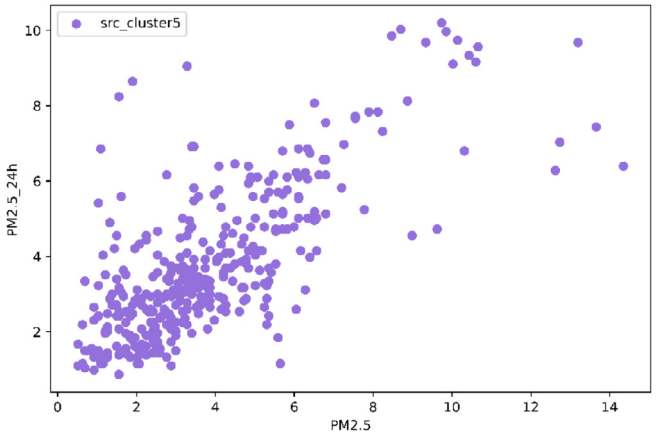

Fig. A.5. The fifth cluster of source domain obtained by GMM.

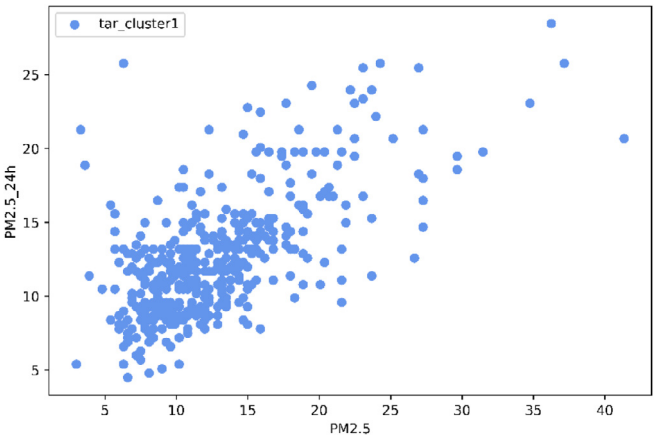

Fig. A.6. The first cluster of target domain obtained by GMM.

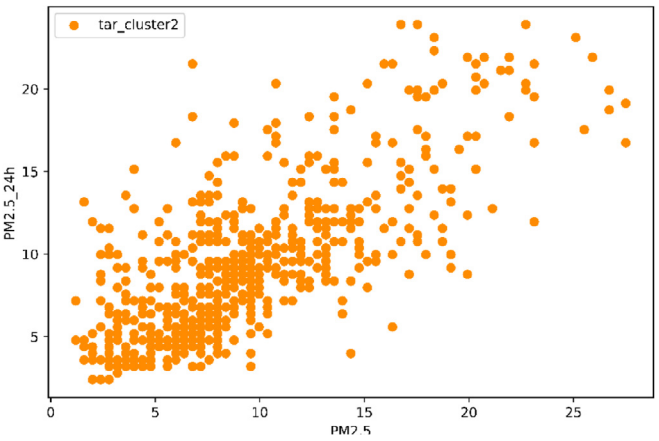

Fig. A.7. The second cluster of target domain obtained by GMM.

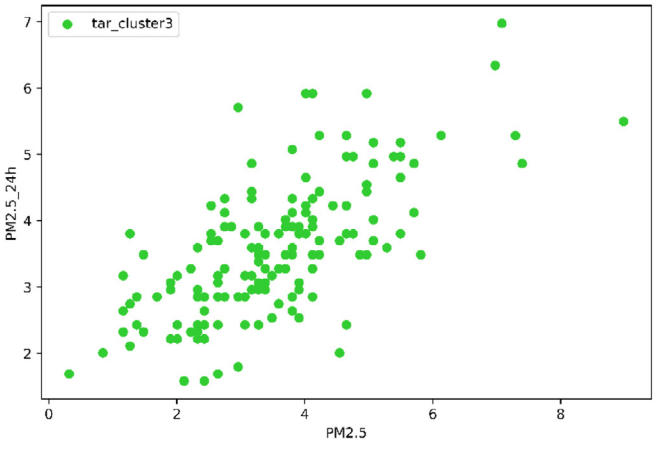

Fig. A.8. The third cluster of target domain obtained by GMM.

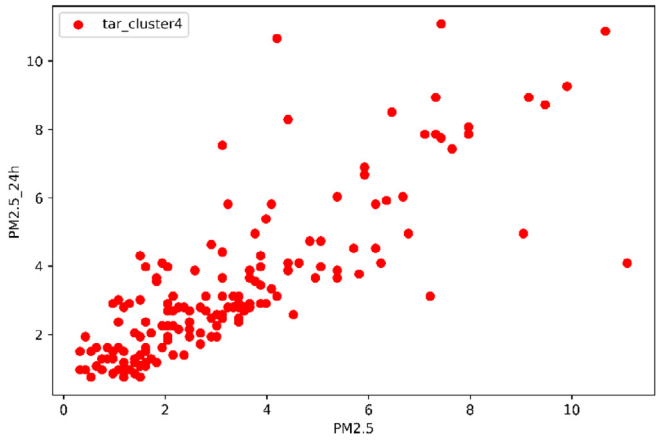

Fig. A.9. The fourth cluster of target domain obtained by GMM.

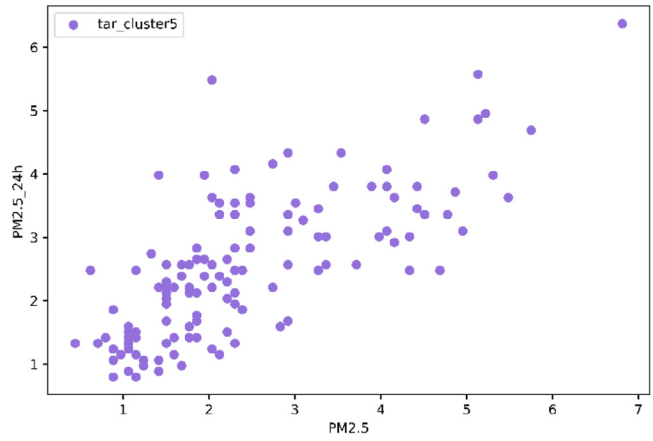

Fig. A.10. The fifth cluster of target domain obtained by GMM.

References

[1] C. Sohrabi, Z. Alsafi, N. O'neill, M. Khan, A. Kerwan, A. Al-Jabir, C. Iosifidis, R. Agha, World health organization declares global emergency: A review of the 2019 novel coronavirus (COVID-19), Int. J. Surg. 76 (2020) 71–76, http://dx.doi.org/10.1016/j.ijsu.2020.02.034.

[2] Y. Bai, L. Yao, T. Wei, F. Tian, D.-Y. Jin, L. Chen, M. Wang, Presumed asymptomatic carrier transmission of COVID-19, JAMA 323 (2020) 1406–1407, http://dx.doi.org/10.1001/jama.2020.2565.

[3] J.D. Berman, K. Ebisu, Changes in U.S. air pollution during the COVID-19 pandemic, Sci. Total Environ. 739 (2020) 139864, http://dx.doi.org/10.1016/j.scitotenv.2020.139864.

[4] D. Fattorini, F. Regoli, Role of the chronic air pollution levels in the COVID-19 outbreak risk in Italy, Environ. Pollut. 264 (2020) 114732, http://dx.doi.org/10.1016/j.envpol.2020.114732.

[5] Y. Huang, J.J.-C. Ying, V.S. Tseng, Spatio-attention embedded recurrent neural network for air quality prediction, Knowl.-Based Syst. 233 (2021) 107416, http://dx.doi.org/10.1016/j.knosys.2021.107416.

[6] X. Li, L. Peng, X. Yao, S. Cui, Y. Hu, C. You, T. Chi, Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation, Environ. Pollut. 231 (2017) 997–1004, http://dx.doi.org/10.1016/j.envpol.2017.08.114.

[7] P.E. Saide, G.R. Carmichael, S.N. Spak, L. Gallardo, A.E. Osses, M.A. Mena-Carrasco, M. Pagowski, Forecasting urban PM10 and PM2.5 pollution episodes in very stable nocturnal conditions and complex terrain using WRF–chem CO tracer model, Atmos. Environ. 45 (2011) 2769–2780, http://dx.doi.org/10.1016/j.atmosenv.2011.02.001.

[8] L.K. Kwok, Y.F. Lam, C.Y. Tam, Developing a statistical based approach for predicting local air quality in complex terrain area, Atmos. Pollut. Res. 8 (2017) 114–126, http://dx.doi.org/10.1016/j.apr.2016.08.001.

[9] J. Ma, J.C.P. Cheng, C. Lin, Y. Tan, J. Zhang, Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques, Atmos. Environ. 214 (2019) 116885, http://dx.doi.org/10.1016/j.atmosenv.2019.116885.

[10] A. Suleiman, M.R. Tight, A.D. Quinn, Applying machine learning methods in managing urban concentrations of traffic-related particulate matter (PM10 and PM2.5), Atmos. Pollut. Res. 10 (2019) 134–144, http://dx.doi.org/10.1016/j.apr.2018.07.001.

[11] S. Chen, J. q. Wang, H. y. Zhang, A hybrid PSO-SVM model based on clustering algorithm for short-term atmospheric pollutant concentration forecasting, Technol. Forecast. Soc. Change 146 (2019) 41–54, http://dx.doi.org/10.1016/j.techfore.2019.05.015.

[12] S. Muhammad, X. Long, M. Salman, COVID-19 pandemic and environmental pollution: A blessing in disguise? Sci. Total Environ. 728 (2020) 138820, http://dx.doi.org/10.1016/j.scitotenv.2020.138820.

[13] J. Ma, J.C.P. Cheng, Y. Ding, C. Lin, F. Jiang, M. Wang, C. Zhai, Transfer learning for long-interval consecutive missing values imputation without external features in air pollution time series, Adv. Eng. Inform. 44 (2020) 101092, http://dx.doi.org/10.1016/j.aei.2020.101092.

[14] P. Zhao, T. Wu, S. Zhao, H. Liu, Robust transfer learning based on geometric mean metric learning, Knowl.-Based Syst. 227 (2021) 107227, http://dx.doi.org/10.1016/j.knosys.2021.107227.

[15] J. Ma, Z. Li, J.C. Cheng, Y. Ding, C. Lin, Z. Xu, Air quality prediction at new stations using spatially transferred bi-directional long short-term memory network, Sci. Total Environ. 705 (2020) 135771, http://dx.doi.org/10.1016/j.scitotenv.2019.135771.

[16] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, G. Zhang, Transfer learning using computational intelligence: A survey, Knowl.-Based Syst. 80 (2015) 14–23, http://dx.doi.org/10.1016/j.knosys.2015.01.010.

[17] B. Liu, C. Liu, Y. Xiao, L. Liu, W. Li, X. Chen, AdaBoost-based transfer learning method for positive and unlabelled learning problem, Knowl.-Based Syst. 241 (2022) 108162, http://dx.doi.org/10.1016/j.knosys.2022.108162.

[18] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (2009) 1345–1359, http://dx.doi.org/10.1109/TKDE.2009.191.

[19] D. Lin, X. An, J. Zhang, Double-bootstrapping source data selection for instance-based transfer learning, Pattern Recognit. Lett. 34 (2013) 1279–1285, http://dx.doi.org/10.1016/j.patrec.2013.04.012.

[20] J. Kim, J. Lee, Instance-based transfer learning method via modified domain-adversarial neural network with influence function: Applications to design metamodeling and fault diagnosis, Appl. Soft Comput. 123 (2022) 108934, http://dx.doi.org/10.1016/j.asoc.2022.108934.

[21] Q.-Q. He, S.W.I. Siu, Y.-W. Si, Instance-based deep transfer learning with attention for stock movement prediction, Appl. Intell. (2022) http://dx.doi.org/10.1007/s10489-022-03755-2.

[22] Q. Chen, B. Xue, M. Zhang, Instance based transfer learning for genetic programming for symbolic regression, in: IEEE Congress on Evolutionary Computation, 2019, pp. 3006–3013, http://dx.doi.org/10.1109/CEC.2019.8790217.

[23] J. Zhao, S. Shetty, J.W. Pan, Feature-based transfer learning for network security, in: IEEE Military Communications Conference, 2017, pp. 17–22, http://dx.doi.org/10.1109/MILCOM.2017.8170749.

[24] A. Farahani, B. Pourshojae, K. Rasheed, H.R. Arabnia, A concise review of transfer learning, in: International Conference on Computational Science and Computational Intelligence, 2020, pp. 344–351, http://dx.doi.org/10.1109/CSCI51800.2020.00065.

[25] S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain adaptation via transfer component analysis, IEEE Trans. Neural Netw. 22 (2010) 199–210, http://dx.doi.org/10.1109/TNN.2010.2091281.

[26] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1717–1724.

[27] F. Li, S.J. Pan, O. Jin, Q. Yang, X. Zhu, Cross-domain co-extraction of sentiment and topic lexicons, in: The 50th Annual Meeting of the Association for Computational Linguistics, Vol. 1, 2012, pp. 410–419.

[28] A. Singhal, P. Singh, B. Lall, S.D. Joshi, Modeling and prediction of COVID-19 pandemic using Gaussian mixture model, Chaos Solitons Fractals 138 (2020) 110023, http://dx.doi.org/10.1016/j.chaos.2020.110023.

[29] H. Zuo, J. Lu, G. Zhang, F. Liu, Fuzzy transfer learning using an infinite Gaussian mixture model and active learning, IEEE Trans. Fuzzy Syst. 27 (2018) 291–303, http://dx.doi.org/10.1109/TFUZZ.2018.2857725.

[30] R. Wang, S. Han, J. Zhou, Y. Chen, L. Wang, T. Du, K. Ji, Y.O. Zhao, K. Zhang, Transfer-learning-based Gaussian mixture model for distributed clustering, IEEE Trans. Cybern. (2022) 1–13, http://dx.doi.org/10.1109/TCYB.2022.3177242.

[31] J. Qiao, X. Cai, Q. Xiao, Z. Chen, P. Kulkarni, C. Ferris, S. Kamarthi, S. Sridhar, Data on MRI brain lesion segmentation using K-means and Gaussian

mixture model-expectation maximization, Data Brief 27 (2019) 104628, http://dx.doi.org/10.1016/j.dib.2019.104628.
[32] B. Sun, J. Feng, K. Saenko, Correlation alignment for unsupervised domain adaptation, in: G. Csurka (Ed.), Domain Adaptation in Computer Vision Applications, Springer International Publishing, Cham, 2017, pp. 153–171.
[33] H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, W. Zuo, Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation, in: IEEE Conference on Computer Vision and Pattern Rrecognition, 2017, pp. 2272–2281.
[34] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, A. Smola, A kernel method for the two-sample-problem, Adv. Neural Inf. Process. Syst. 19 (2006) http://dx.doi.org/10.48550/arXiv.0805.2368.
[35] S.M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: Advances in Neural Information Processing Systems, Vol. 30, 2017, http://dx.doi.org/10.48550/arXiv.1705.07874.
[36] A.B. Parsa, A. Movahedi, H. Taghipour, S. Derrible, A. Mohammadian, Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis, Accid. Anal. Prev. 136 (2020) 105405, http://dx.doi.org/10.1016/j.aap.2019.105405.
[37] T. Van Nguyen, L. Zhou, A.Y.L. Chong, B. Li, X. Pu, Predicting customer demand for remanufactured products: A data-mining approach, European J. Oper. Res. 281 (2020) 543–558, http://dx.doi.org/10.1016/j.ejor.2019.08.015.
[38] D.M. dos Reis, P. Flach, S. Matwin, G. Batista, Fast unsupervised online drift detection using incremental Kolmogorov-Smirnov test, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1545–1554, http://dx.doi.org/10.1145/2939672.2939836.
[39] A. Chakrabarti, J.K. Ghosh, AIC, BIC and recent advances in model selection, Philos. Stat. 7 (2011) 583–605, http://dx.doi.org/10.1016/B978-0-444-51862-0.50018-6.
[40] S. Rath, A. Tripathy, A.R. Tripathy, Prediction of new active cases of coronavirus disease (COVID-19) pandemic using multiple linear regression model, Diabet. Metabol. Syndrome: Clin. Res. Rev. 14 (2020) 1467–1474, http://dx.doi.org/10.1016/j.dsx.2020.07.045.
[41] M. Saqib, Forecasting COVID-19 outbreak progression using hybrid polynomial-Bayesian ridge regression model, Appl. Intell. 51 (2021) 2703–2713, http://dx.doi.org/10.1007/s10489-020-01942-7.
[42] R. Tibshirani, Regression shrinkage and selection via the Lasso, J. R. Stat. Soc. Ser. B Stat. Methodol. 58 (1996) 267–288, http://dx.doi.org/10.1111/j.2517-6161.1996.tb02080.x.
[43] F. Amini, G. Hu, A two-layer feature selection method using genetic algorithm and elastic net, Expert Syst. Appl. 166 (2021) 114072, http://dx.doi.org/10.1016/j.eswa.2020.114072.
[44] F.-K. Wang, T. Mamo, Gradient boosted regression model for the degradation analysis of prismatic cells, Comput. Ind. Eng. 144 (2020) 106494, http://dx.doi.org/10.1016/j.cie.2020.106494.
[45] M. Klug, Y. Barash, S. Bechler, Y.S. Resheff, T. Tron, A. Ironi, S. Soffer, E. Zimlichman, E. Klang, A gradient boosting machine learning model for predicting early mortality in the emergency department triage: Devising a nine-point triage score, J. Gen. Intern. Med. 35 (2020) 220–227, http://dx.doi.org/10.1007/s11606-019-05512-7.
[46] P. Chou, H.H.-C. Chuang, Y.-C. Chou, T.-P. Liang, Predictive analytics for customer repurchase: Interdisciplinary integration of buy till you die modeling and machine learning, European J. Oper. Res. 296 (2022) 635–651, http://dx.doi.org/10.1016/j.ejor.2021.04.021.
[47] B. Sun, K. Saenko, Deep coral: Correlation alignment for deep domain adaptation, in: European Conference on Computer Vision, 2016, pp. 443–450, http://dx.doi.org/10.1007/978-3-319-49409-8_35.
[48] J. Wang, Y. Chen, S. Hao, W. Feng, Z. Shen, Balanced distribution adaptation for transfer learning, in: IEEE International Conference on Data Mining, 2017, pp. 1129–1134, http://dx.doi.org/10.1109/ICDM.2017.150.