Results in Engineering 24 (2024) 103290

Contents lists available at ScienceDirect

Results in Engineering

journal homepage: www.sciencedirect.com/journal/results-in-engineering

Research paper

Explainable deep learning on multi-target time series forecasting: An air pollution use case

Manuel J. Jiménez-Navarro [a,*], Mario Lovrić [b,c], Simonas Kecorius [d], Emmanuel Karlo Nyarko [e], María Martínez-Ballesteros [a]

[a] *Department of Computer Languages and Systems, University of Seville, ES-41012 Seville, Spain*
[b] *Centre for Bioanthropology, Institute for Anthropological Research, Zagreb, Croatia*
[c] *The Lisbon Council, Brussels, Belgium*
[d] *Institute of Epidemiology, Helmholtz Zentrum München - German Research Center for Environmental Health, Neuherberg, Germany*
[e] *Faculty of Electrical Engineering, Computer Science and Information Technology Osijek, Josip Juraj Strossmayer University of Osijek, Osijek, Croatia*

ARTICLE INFO

Keywords:
Time series forecasting
Air pollution
Feature selection
Deep learning
XAI

ABSTRACT

Urban air pollution represents a significant threat to public health and the environment, with nitrogen oxides, ozone, and particulate matter being among the most harmful pollutants. These contribute to respiratory and cardiovascular diseases, particularly in urban areas with high traffic and elevated temperatures. Machine learning, especially deep learning, shows promise in enhancing the prediction accuracy of prediction of pollutant's concentrations. However, the "black box" nature of these models often limits their interpretability, which is crucial for informed decision-making. Our study introduces a Temporal Selection Layer technique within deep learning models for time series forecasting to tackle this issue. This technique not only improves prediction accuracy by embedding feature selection directly into the neural network, but also enhances interpretability and reduces computational costs. In particular, we applied this method to hourly concentration data of pollutants, including particulate matter, ozone, and nitrogen oxides, from five urban monitoring sites in Graz, Austria. These concentrations were used as target variables to predict, while identifying the most relevant features and periods that affect prediction accuracy. Comparative analysis with other embedded feature selection methods showed that the Temporal Selection Layer significantly enhances both model effectiveness and transparency. Additionally, we applied explainable techniques to evaluate the impact of weather and time-related factors on air pollution, which also helped assess feature importance. The results show that our approach improves both prediction accuracy and model interpretability, leading finally to more effective pollution management strategies.

1. Introduction

Ambient air quality has become a pressing environmental concerns of our time, with significant implications for human health [1], ecosystems [2], and global climate [3]. Among the multitude of pollutants that contaminate the atmosphere, nitrogen oxides (NO and NO_2), ozone (O_3), and particulate matter (PM) stand out due to their pervasive presence and severe health effects. Nitrogen oxides, primarily generated by vehicles and industrial processes [4], are key precursors in the formation of ground-level ozone, a potent pollutant that forms through complex photochemical reactions involving volatile organic compounds under sunlight [5]. This process is particularly intensified in urban environments, where high traffic density and increased temperatures create ideal conditions for ozone accumulation [6].

Particulate matter is another major concern, especially $PM_{2.5}$ (particulate matter with a diameter below 2.5 μm) and PM_{10} (particulate matter with a diameter below 10 μm), which differ in size but share the common trait of being small enough to be inhaled into the human respiratory system [7]. $PM_{2.5}$ poses a risk as it can penetrate into the lungs, leading to serious health conditions. Recent studies have also shown that exposure to $PM_{2.5}$ is correlated with an increased risk of developing various mental disorders, including depression and anxiety, highlighting the broader systemic risks associated with fine particulate matter exposure [8]. On the other hand, PM_{10} particles, while larger, are still

[*] Corresponding author.
E-mail addresses: mjimenez3@us.es (M.J. Jiménez-Navarro), mario.lovric@inantro.hr (M. Lovrić), simonas.kecorius@helmholtz-muenchen.de (S. Kecorius), karlo.nyarko@ferit.hr (E.K. Nyarko), mariamartinez@us.es (M. Martínez-Ballesteros).

https://doi.org/10.1016/j.rineng.2024.103290
Received 14 September 2024; Received in revised form 22 October 2024; Accepted 30 October 2024

Available online 6 November 2024

harmful and can cause respiratory issues and irritation of the eyes, nose, and throat. In certain industrial regions, exposure to PM_{10} has been associated with elevated levels of heavy metals, which pose significant health risks, especially to vulnerable populations such as children [9].

In the case of O_3, while essential in the stratosphere for protecting the Earth from harmful ultraviolet radiation, becomes a dangerous pollutant at ground level. It not only aggravates respiratory conditions, but also plays a significant role in climate warming due to its greenhouse gas properties [6]. The significant impact of these pollutants on public health is evident, with the World Health Organization estimating that it is responsible for approximately 4.2 million premature deaths each year, an alarming statistic that underscores the magnitude of this issue [10].

In response to the growing concern over air quality, there is a need to model their ambient concentrations for monitoring purposes, understanding sources, short- and long-term trends [11] and mutual relationships [12]. Machine learning algorithms came in handy due to their data-driven nature, while first-principle models still serve to understand the spatial distributions of pollutants. Despite their effectiveness in improving prediction accuracy, deep learning (DL) models often face significant challenges related to their lack of interpretability, which is crucial for informed decision-making [13]. Building trust, enhancing the understanding of decision-making processes, and ensuring accountability in high-stakes applications are critical challenges that eXplainable Artificial Intelligence (XAI) addresses effectively [14].

To address this issue, our study applies a recent technique referred to as Temporal Selection Layer (TSL) within DL models such as Feed-Forward (FF) and Long-Short-Term Memory (LSTM) for time series forecasting. This technique not only improves prediction accuracy by embedding feature selection directly into the neural network but also reduces computational complexity and enhances model interpretability [15]. Because this process is embedded within the model, it has an additional property by offering insights into the features deemed relevant and irrelevant. This strategy improves the interpretation of the model reducing their black box nature.

Our main contribution focuses on predicting the airborne particle concentration using data from urban monitoring stations, with the aim of enhancing prediction precision and identifying the most relevant features and periods that contribute to prediction accuracy. Specifically, we applied the proposed methodology to public data from Austrian government sources [16], covering the period from January 2014 to March 2022. In particular, the data collected includes measurements of PM_{10}, NO, and NO_2 in five urban areas of Graz, Austria.

Additionally, we compared the performance of standard deep learning models FF and LSTM, both with and without the embedded TSL. These methods are compared against classical techniques such as Decision Tree (DT) Lasso (L1), K-Nearest Neighbors (KNN), and eXtreme Gradient Boosting (XGB) models. To assess the importance of features in the best-performing model, we also applied the well-known XAI technique named SHapley Additive exPlanation (SHAP) [17], which provide valuable insights into how past endogenous and exogenous features influence the predictions. SHAP assigns a numerical importance score to each input variable that contributes to the model's prediction, based on Shapley values from game theory [18]. This technique has been successfully applied in several domains, including agriculture, cybersecurity, healthcare, finance, and natural language processing, among others, to enhance model transparency and interoperability [19,20].

The structure of this paper is as follows: Section 2 examines the latest developments in the explainability of deep learning models, particularly in the context of air pollution prediction. Section 3 details the materials and methods used in this research, covering the model fundamentals, TSL and SHAP techniques, as well as the input data and the metrics used for model comparison. It also provides an overview of the hyperparameter space and the optimization process. Section 4 outlines and analyzes the primary findings from the experiments. Lastly, Section 5 summarizes the conclusions of the study and outlines potential directions for future research.

2. Related work

Several studies have applied machine learning to assess environmental impacts. For example, the authors in [21] used random forest regression to assess air quality changes during the COVID-19 lockdown, finding significant reductions in NO_2 and PM_{10} due to decreased traffic, while O_3 levels increased. This highlights the effectiveness of predictive models in analyzing environmental changes during reduced economic activity. Similarly, [22] demonstrates the effectiveness of 1D convolutional neural networks in modeling daily airborne particle concentrations, highlighting the potential of deep learning in environmental monitoring.

The increasing complexity of machine learning models used in environmental forecasting, particularly deep learning models, has underscored the need for greater transparency and interpretability. Recent work has begun to explore the role of XAI techniques in improving the transparency and explainability of these models. For instance, [23] used the Extra Trees model to predict ground-level ozone and provided an interpretable rule-based framework, identifying key factors such as air temperature and pollutant concentrations.

These examples show that XAI techniques are crucial in making black-box models more understandable, particularly in the context of time series forecasting.

Explainability methods in deep learning and multivariate time series forecasting can generally be categorized into two main families: Ante-hoc and Post-hoc [24].

- The Ante-hoc family focuses on model interpretability, providing insights into the internal workings of the models to explain the behavior learned during training. Over the years, several types of Ante-hoc methods have emerged, including approaches based on feature importance, decision rules, and time series decomposition.
  - Feature importance methods offer a numerical or binary representation of a feature's relevance at either a local or global level. Examples of these methods include tree-based models [25], attention-based deep learning architectures [26], and embedded feature selection techniques [27] among others.
  - Decision rules methods extract the conditions that lead to a final decision, similar to the way decision tree models make predictions. Some examples are the GRU-Tree model [28] and Neuro-fuzzy approaches [29].
  - Decomposition methods decompose the original time series into its primary components, such as trend, seasonality, and residuals. The N-BEATS architecture [30] is a common example in this category.
- The Post-hoc family explains models using XAI techniques after the model has been trained. In time series forecasting, the most common Post-hoc methods are based on feature importance. Notable examples include SHAP [31], GRAD-CAM [32], LIME [33], and association rules [34].

Feature selection has become a critical aspect of XAI, as it enhances both the interpretability and performance of machine learning models [35,36]. Embedded feature selection methods, in particular, integrate the selection process within the model training, leading to more efficient models that are both more transparent and better performing. These approaches not only enhance the interpretability of the model but also reduce computational costs.

In recent years, time series forecasting models have achieved notable success in various domains [37]. However, as these models are increasingly applied to more complex and critical problems, the demand for accuracy and interpretability has grown. This has underscored the importance of feature selection and XAI techniques [38], which are

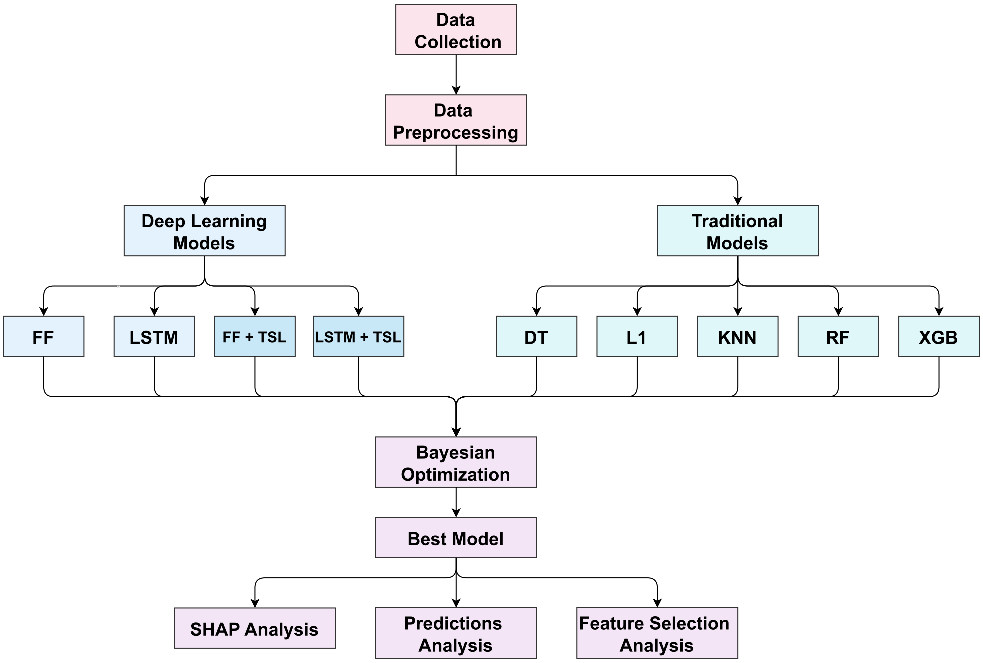

Fig. 1. Flowchart depicting the sequence of steps in the proposed experimental framework.

essential to enhance transparency and understanding in temporal data analysis. By focusing on the most relevant features, especially within the temporal dimension, these methods ensure that the models remain efficient and transparent, leading to more understandable and reliable predictions [39].

Several studies have specifically investigated their use in analyzing air pollutant levels. For example, [12,21] applied permutation importance was applied to explain feature selection during model training. However, this approach has limitations, as it can overlook important features and combinations of features.

In response to these challenges, the TSL methodology was introduced as an embedded feature selection approach to improve deep learning model interpretability. Tested on several classification and regression datasets from fields like Health, Economics, and Environment, TSL demonstrated notable improvements over traditional feature selection techniques, offering reliable and efficient results [38]. TSL outperformed baseline models and filter-based methods across most datasets with simpler parametrization and less information [40]. Applied to ozone level prediction in Spain, TSL improved model accuracy and interpretability, achieving a 9% effectiveness increase compared to models such as Lasso and Decision Tree [15].

Based on the review of the existing literature, it is clear that the integration of feature selection techniques and XAI methods has become increasingly important in the field of environmental modeling. As demonstrated by recent studies, embedded feature selection methods not only enhance model transparency but also improve their predictive capabilities. In this work, we focus specifically on these techniques to further develop interpretable and efficient models for air quality forecasting.

3. Materials and methods

This section details the materials and methods used in this study, beginning with a comprehensive description of the data sources and the preprocessing steps necessary for effective analysis. It also presents the model selected for comparative analysis, highlighting the key features of TSL and the post hoc explainability technique used to improve both performance and interpretability. In addition, this section outlines the evaluation metrics to assess the effectiveness of the models. Fig. 1 illustrates the flowchart followed during the experimentation.

3.1. Data description

A long term (almost nine years) environmental, pollution, and weather data set from 5 measuring stations from the Austrian city of Graz (Austria) namely Sud, Nord, West, Ost, and Don Bosco was processed and deposited at [41]. The measurements, taken hourly, covered the time period from January 2014 to March 2022. The stations have been described in [21] and [22]. The concentrations of NO_2, NO, O_3, and PM_{10} were compiled into a table alongside with meteorological variables from ground-level measurements from the different stations. Refer to the original paper for a complete list of ground-level meteorological features.

This study extends previous research by incorporating ERA5 reanalysis data, in addition to these features. ERA5-Land is a global meteorological dataset covering the period from 1950 to the present with a $0.1^\circ \times 0.1^\circ$ resolution [42,43]. ERA5-Land refines simulated land fields and ERA5 atmospheric variables such as air temperature and humidity. The data was retrieved through the Copernicus Climate Data Store API and aggregated into a daily frequency. Table 1 contains a complete list of the features used. For further information, please refer to the original ERA5 source [44]. The features have been calculated and presented in [45].

Understanding pollution levels in advance is crucial for planning and taking action during periods of higher concentrations. Therefore, the problem requires a minimum of a 24-hour forecast that provides hourly pollution levels, with predictions made at the start of each day.

A Bayesian Ridge model was employed to impute missing values, treating each feature as a function of the others in a round-robin manner. Once the missing values were filled, the lag features were generated using meteorological data from all five stations. This involved using a 12-hour window, where the median of the 12 most recent measurements was calculated to represent the value for each hour [12]. To avoid ambiguity between 0° and 360°, wind direction was expressed in terms of x and y coordinates. Wind speed at a height of 10 meters was derived from its u and v components.

Table 1
Table of variables and their measuring units.

Feature	Abbreviation	Unit
10m u-component of wind	u10	m s^{-1}
10m v-component of wind	v10	m s^{-1}
2m temperature	d2m	K
Soil temperature level 4	stl4	K
Snow cover	snowc	%
Snow depth	sd	m
Snow depth of water equivalent	sde	m of water equivalent
Snowfall	sf	m of water equivalent
Snowmelt	smlt	m of water equivalent
Temperature of snow layer	tsn	K
Forecast albedo	fal	dimensionless
Surface net solar radiation	rsn	J m^{-2}
Surface sensible heat flux	slhf	J m^{-2}
Surface thermal radiation downwards	strd	J m^{-2}
Surface latent heat flux	slhf	J m^{-2}
Surface net thermal radiation	strd	J m^{-2}
Surface latent heat flux	sshf	J m^{-2}
Total precipitation	tp	M
Windspeed	ws	m s^{-1}

3.2. Models

In this study, a diverse set of models was been used to evaluate their performance in the problem at hand. These models belong to the most commonly studied family of methods in the literature for multi-horizon time series forecasting of air pollution, chosen for their effectiveness and efficiency. The model families considered include: linear, lazy, tree-based, ensemble, and neural networks. Given the high dimensionality of the problem, interpreting the relevant features is crucial. For this reason, a representative model from each family is selected, with embedded feature selection methods included where applicable.

The Decision Tree (DT) [46] model represents the tree-based family. It is widely used and effective, having demonstrated success in a wide range of problems.

The Lasso (L1) [47] model represents the linear family. Known for its embedded feature selection through regularization, Lasso has been extensively studied and shown to perform well in various forecasting tasks.

For the lazy family, the Nearest Neighbors (KNN) [48] model is chosen due to its simplicity and efficiency in terms of training time. However, a limitation of KNN is the absence of an embedded feature selection mechanism.

Ensemble models cover a wide range of techniques. In this study, two of the most common ensemble methods from the literature are chosen. Random Forest (RF) [49] represents the bagging approach and is one of the most effective methods in machine learning. Extreme Gradient Boosting (XGB) [50] represents the boosting approach and provides state-of-the-art results for many tabular data problems. However, neither RF nor XGB offers embedded feature selection and both lack interpretability.

Two different neural network architectures are also selected: Feed-forward (FF) [51] and Long Short-Term Memory (LSTM) [52]. Both have shown a strong performance in time series forecasting problems [53,54]. However, like ensemble models, neural networks do not inherently offer interpretability or feature selection.

To address this issue, each neural network models are enhanced by incorporating a Time Selection Layer (TSL) [15] resulting in TFF and TLSTM models. TSL is an embedded feature selection method that improves the interpretability and predictive performance of these models by filtering out irrelevant features, including those in the temporal domain. It achieves this by performing element-wise multiplication between binarized weights and the input tensor, effectively excluding features that do not contribute to the prediction. Positioned at the top of the network architecture, this layer is formally defined as illustrated in the Equation (1):

$$TSL(X^{MxD}) = H(W^{MxD}) \circ X^{MxD} \tag{1}$$

In this scenario, "∘" represents the Hadamard product. Here, W stands for the weight matrix linked to historical data and D represents the count of features. H functions as a gate, determining feature selection by approximating the Heaviside step function.

TSL is integrated into the neural network and is trained through backpropagation, and during training, features with zero weights in the selection mask are automatically excluded. Lasso regression is applied as regularization to avoid removing important features, penalizing the number of selected features based on the number of forecasting steps, thereby enhancing both model performance and transparency.

3.3. Model evaluation

In this study, we assess the performance of the models using a set of widely recognized metrics in forecasting tasks in the context of air pollution. These metrics include the Mean Absolute Error (MAE), which is the most common metric in forecasting problems, Root Mean Squared Error (RMSE), the most common metric in air pollution contexts, and Weighted Absolute Percentage Error (WAPE), which provides an interpretable metric in percentage measure (range [0, 1]). Each metric offers a different perspective on the error obtained by each model which would help to understand their strengths and weaknesses.

In addition, the required time to fit the different models has been measured in terms of seconds. This time will provide insights about the ability of the models to scale to high-dimensional problems. Note that due to the early stopping technique applied during the training, this time measures the convergence time which has no relationship with the number of parameters present in the model.

Evaluating a model properly requires providing the most reliable metrics that measure not only its efficacy but also its ability to adapt to unknown scenarios. Therefore, a Blocked Cross-Validation (BCV) approach is employed with four different folds, each containing data from six consecutive years. For instance, the first block is composed of the following splits: years [2014, ..., 2017] for training, year 2018 for validation, and year 2019 as testing. Note that as we use the BCV, the size of every split is always the same. Thus, the second fold would contain: years [2015, ..., 2018] for training split, year 2019 for validation split, and year 2020 as testing split. This process is repeated for the next two folds modifying the years contained in every split.

The evaluation is performed over the trained models in each fold, evaluated on the test set. With this approach, we can avoid providing more unbiased metrics that evaluate the generability and robustness of the models.

3.4. Hyperparameter space

The models described in Section 3.2 typically rely on a predefined set of hyperparameters that perform well in a wide range of problems. However, fine-tuning these hyperparameters can significantly improve performance in most cases. In this section, we detail the bounds established for the various models used in our study, which are dictated by computational or technical limitations of the respective libraries.

Table 2 outlines the configuration for each model. The first entry includes the window size hyperparameter, which relates to the preprocessing applied to the input data and is included for clarity. Tree-based methods share the maximum depth hyperparameter, with the same range across models, except for XGB, where the range is constrained by the underlying technology. The L1 model only requires the regularization term (also known as alpha), which is set between zero (no penalization, linear model) and one. For the KNN model, the number of neighbors is the key hyperparameter, with an upper bound of 32 deemed appropriate for this study.

Neural networks require more hyperparameters than other methods but can deliver strong results, as discussed in Section 4. Since very deep

Table 2
Hyperparameter ranges defined during tuning. Note that the first row is an hyperparameter common to every model.

Model	Hyperparameter	Range
-	Window size	[24, 72]
DT	Maximum depth	[2, 32]
L1	Regularization	[0, 1]
	Layers	[1, 3]
	Units	[4, 2048]
FF & LSTM	Dropout	[0, 0.5]
	Learning rate	[0.0001, 0.01]
	Batch size	[16, 128]
TSL	Regularization	[0.01, 1e-7]
KNN	K	[2, 32]
RF	Maximum depth	[3, 32]
	# Estimators	[2, 100]
	Min samples	[2, 256]
XGB	Maximum depth	[1, 16]
	# Estimators	[2, 32]
	Learning rate	[0, 1]
	Min samples	[2, 256]

architectures are not typically necessary for time series forecasting problems, the upper limit was set to three layers. The number of units, capped at 2048, was chosen to be as large as computationally feasible, given the high dimensionality of the input features. Standard ranges were applied for Dropout, learning rate, and batch size, as commonly found in the literature.

Lastly, the range for the TSL layer was selected based on insights from previous studies on similar problems, which helped inform the most suitable parameters.

3.5. Hyperparameter optimization

The models used in this work require specific configurations that significantly influence their final performance. These configurations, often referred to as hyperparameters, determine aspects such as model design, constraints, and the learning process. To identify the optimal configuration, an optimization algorithm is needed to explore the search space and evaluate different setups.

In this study, Bayesian optimization is used to find the best set of hyperparameters for each model due to its proven effectiveness in recent research [55,56]. The key inputs for this algorithm are the objective function to be optimized (discussed in Section 3.3) and the search space (outlined in Section 3.4), both of which have been previously detailed.

Throughout multiple iterations, the chosen model is trained and the algorithm updates the distribution of hyperparameters based on the performance of previous configurations. This update is guided by the average MSE metric, calculated across different validation years (refer to Section 3.3). Once the maximum number of iterations is reached, the configuration that delivers the best performance is selected. In this study, a maximum of 25 iterations was considered, which, with 4 validation folds, resulted in the evaluation of 100 different models. Additionally, the algorithm employed the default Upper Confidence Bound (UCB) acquisition function, with a λ value of 2.576, to balance exploration and exploitation.

3.6. Post-hoc explainability with SHAP

Feature selection provides some highlights about the relevant features which increase the interpretability of the black-box model. However, we cannot quantify the influence of the relevant features on the final predictions. This may be especially important in high dimensional spaces where there may be thousands of relevant selected features. In addition, the relevant features are not linked with any output in specific but with all the different outputs at the same time, which can hinder the interpretation.

Table 3
Best model configuration and metrics averaged for each fold. The best model for each metric is highlighted in bold.

Model	MAE	MSE	RMSE	WAPE	Time (min)
DT	11.942	357.718	18.769	0.483	31.176
FF	11.340	330.139	18.048	0.455	4.735
KNN	12.472	380.953	19.118	0.497	**0.007**
L1	12.149	506.006	21.242	0.495	2169.395
LSTM	11.269	318.169	17.644	0.450	3.104
TFF	**10.275**	**283.675**	**16.740**	**0.415**	25.653
TLSTM	11.406	315.868	17.592	0.460	6.211
RF	11.008	292.894	16.993	0.444	2317.936
XGB	17.395	592.307	24.312	0.711	36.840

For a deeper understanding, we applied SHAP [17], a post-hoc explainability technique grounded in cooperative game theory. This technique aims to provide more information, indicating the level of influence of each feature on the prediction. SHAP assigns a Shapley value [18] to each feature, which quantifies its average marginal contribution to the model's predictions. This method not only identifies the relevant features but also provides insights into their global importance, thereby enriching the interpretability of the model.

Due to the post-hoc nature, the explanation process starts from the trained model and an analysis dataset. The analysis dataset is obtained by selecting the 20 centroids from the K-Means algorithm to select the most representative instances. After that, as the SHAP technique employs a local explanation approach, the relevance is calculated for each dataset instance. Finally, the relevance is averaged for each instance to obtain the final relevance for each feature.

In the case of models that embed a feature selection process, additional post-processing must be applied to obtain accurate relevance values. One flaw of the SHAP technique is that its importance values can be misleading when two features are correlated. For that reason, removed features can have a non-zero Shapley value if correlated with a relevant feature, which is likely to happen in time series data. For that reason, the Shapley value for removed features is set to zero for the models that embed a feature selection process.

4. Experimental results and discussion

In this section, the main results are described after the experimentation performed. These results are divided into several sections to analyze the performance from multiple perspectives using a general-to-specific structure. Firstly, Section 4.1 presents the results obtained for each model using the best configuration found. Section 4.2 details the configuration for each model presented in previous section. The error is specified and divided by each target in Section 4.3. The best model found starts to be explained in Section 4.4 describing the selected features. Section 4.5 quantifies the relevance of each feature on the predictions. Finally, Section 4.6 shows graphically the predictions performed by the model compared with the real evolution.

4.1. Best results

The efficacy obtained in terms of our selected evaluation metrics is reported in Table 3. Each row represents a model acronym, while each column represents a different metric. The metric values represent the average values for each predicted target in the testing fold, using the same model configuration. Additionally, the table includes the average training time in minutes for the complete four-fold experiment.

In general, the error metrics highlight the complex, non-linear nature of the problem. Simpler models like L1, KNN, or DT exhibit 3% to 8% more error compared to the more complex models considered in our study. This increased error may be due to the models' inability to adequately adapt to the varying distribution of targets in the multi-step forecasting problem. An exception is the XGB model, which, despite being considered a complex model, produced the worst errors overall. This

Table 4
RMSE metric obtained for each model over the tested years. The best model by year is highlighted in bold.

Year	DT	FF	KNN	L1	LSTM	TFF	TLSTM	RF	XGB
2019	18.5	16.8	17.6	**15.1**	16.6	16.0	16.1	16.1	24.9
2020	17.1	16.2	16.7	16.0	15.3	15.1	**14.9**	15.6	23.3
2021	16.8	17.6	16.2	33.6	16.6	15.9	17.7	**15.8**	23.2
2022	22.6	21.6	25.9	20.3	22.1	**19.9**	21.6	20.5	25.8

could be a consequence of the GPU configuration used during the experimentation. Notably, the XGB library defined different methods that may have sacrificed precision for greater acceleration.

Focusing on complex models, we can distinguish three groups: neural networks, neural networks with TSL, and RF. The results for FF and LSTM architectures are quite similar, with a near 0.5% error difference in terms of WAPE. The inclusion of TSL in the architectures does not seem to have a positive effect on the LSTM model and even deteriorates its efficacy. However, adding TSL to the FF model results in a significant improvement, averaging a 4% reduction in error metrics. RF surpasses every model except the TFF model, closely matching the performance of the simple LSTM model. Nevertheless, there is a clear distinction between the TFF and all other models, and the TFF model is the best in terms of error reduction.

In terms of efficiency, the high dimensionality of the problem posed a significant scalability challenge for the different models. Models with poor scalability experienced substantial increases in training time despite their simplicity. KNN achieved the best efficiency, as expected, because its algorithm does not require a training process, unlike other models. Tree-based models experienced a large increase in training time, especially the random forest, which was the most inefficient. The L1 model was the second most inefficient, likely due to the complexity of optimizing it using the coordinate descent algorithm. Finally, neural networks demonstrated excellent scalability for this problem. However, the inclusion of TSL considerably increased the training time.

After studying the general results averaged by year, Table 4 breaks down this average into its components. The rows represent the different testing years, and the columns represent the various models. The cell values indicate the RMSE metric.

There is a significant increase in error between the first three tested years and the last one, supporting the idea of a substantial concept drift in 2022. This phenomenon helps to evaluate the models' generalizability and robustness, rather than just their memorization ability. No recognizable pattern has been found in previous years, suggesting a gradual concept drift over time.

As presented in Table 3, the best models are neural networks and random forests. Notably, L1 achieved remarkable results in the first two years, potentially establishing it as one of the top methods. However, its lack of generalization led to a significant error increase in 2021. Our study evaluates not only the plain effectiveness of the models but also their robustness over different years. A method that provides notable results in specific scenarios is not suitable for real-world applications. A robust, generalizable, effective, and efficient method is the desirable solution.

Focusing on TFF and RF, the RMSE shows minimal differences in 2019 and 2021, with only a 0.1 difference. The improvement in 2020 and 2022 sets the TFF as the best model.

In conclusion, the general metrics evaluated indicate that the TFF model is the best option, offering the optimal balance between efficacy and efficiency. Additionally, the error evolution over the years highlights it as the most robust and generalizable method.

4.2. Best hyperparameters

As a result of the Bayesian optimization process, the best configuration was identified for each model. These configurations correspond to the results presented in the previous section.

Table 5
Best hyperparameter configurations found. Note that W denotes window size.

Model	W	Configuration
DT	84	Max depth: 4
FF	32	# Layers: 3, # Units: 252, Batch size: 112, Learning rate: 0.0015, Dropout rate: 0.3736
KNN	66	K: 29
L1	120	Apha: 0.0274
LSTM	32	# Layers: 3, # Units: 252, Batch size: 112, Learning rate: 0.0015, Dropout rate: 0.3736
RF	124	Max depth: 38, # Estimators: 59, Min samples: 39
XGB	167	Max depth: 1, # Estimators: 22, Learning rate: 0.0, Min samples: 238
TFF	166	# Layers: 4, # Units: 1533, Batch size: 51, Learning rate: 0.0003, Dropout rate: 0.3433, Regularization: 0.0075
TLSTM	94	# Layers: 3, # Units: 113, Batch size: 30, Learning rate: 0.0022, Dropout rate: 0.0097, Regularization: 0.0027

Table 5 presents each model along with its window size (W) and configuration. Generally, most models require a window size exceeding half of the maximum value, indicating that this problem demands a large number of lags to extract relevant information from the dataset.

For tree-based models, the key factor affecting performance was the maximum depth allowed. The poor performance of XGB is explained by its depth of one and a learning rate of zero, leading to underfitting. The decision tree increased its depth to four, which, as previously discussed, was insufficient for good results. The random forest achieved the best results with a maximum depth of 38 decisions, reflecting the complexity of the problem and the large number of features.

The KNN model used 29 neighbors, suggesting underfitting due to excessive smoothing of predictions.

In the L1 model, the only optimized parameter was the alpha regularization factor, which had a low value. This indicates that the model, like the random forest, considers a large number of features.

Interestingly, the neural networks without TSL had almost the maximum allowed layers, although the number of units per layer was significantly below the maximum. The batch size and dropout rate were close to their maximum allowed values, while the learning rate was half the default for the Adam optimizer.

The TSL models, particularly the TFF model, differed notably from other neural networks. The TLSTM model had the same number of layers, emphasizing the problem's complexity, but reduced all other parameters except for the learning rate. Its low regularization term suggests that a large number of features were selected.

The TFF model stood out by using the maximum number of layers and units, indicating its superior performance was due to increased complexity. The best TFF configuration supports the idea that the complexity of this model allows for more effective processing and extraction of useful information with fewer input features. Additionally, its regularization term was nearly three times that of the TLSTM, likely due to enhanced complexity, enabling more refined information extraction with fewer features.

Table 6
Error by horizon for the best model.

Target	MAE	MSE	RMSE	WAPE
Ost|NO	11.182	376.817	19.412	0.771
Ost|NO_2	8.095	114.199	10.686	0.358
Ost|$PM_{10}K$	8.484	188.076	13.714	0.348
West|NO	9.116	276.128	16.617	0.862
West|NO_2	8.260	125.059	11.183	0.372
West|$PM_{10}K$	7.342	116.435	10.790	0.358
Nord|O_3	16.944	471.407	21.712	0.382
Nord|NO	5.858	134.276	11.588	0.987
Nord|NO_2	7.081	94.468	9.719	0.394
Nord|$PM_{10}K$	6.657	92.502	9.618	0.352
Sud|O_3	17.331	487.797	22.086	0.455
Sud|NO	15.670	840.574	28.993	0.770
Sud|NO_2	8.301	125.539	11.204	0.350
Sud|$PM_{10}K$	7.997	142.005	11.917	0.353
DonBosco|NO	19.620	1035.696	32.182	0.573
DonBosco|NO_2	9.256	148.446	12.184	0.272
DonBosco|$PM_{10}K$	8.372	147.885	12.161	0.344

4.3. Target error

To better understand the predictions made by the best model, the error metrics for each target were analyzed. These metrics were calculated by averaging the errors across all forecasting horizons within each test set.

Table 6 summarizes the errors for each target (y-axis) and metric (x-axis). The error distribution across different pollutants varies significantly by area, probably due to spatial differences in pollutant characteristics.

The NO pollutant shows the most variation, with a minimum MAE of 5.9 in the Nord and a maximum in Don Bosco. This variation could be linked to differences in population density and traffic flow, leading to less variability in the data.

NO_2, O_3, and $PM_{10}K$ pollutants exhibit less deviation than NO. However, a similar pattern emerges, with the Nord showing an MAE difference of one to two points compared to other areas.

4.4. TSL selected features

Following a general-to-specific structure, we selected the TFF model as the best overall performer. In this and the following sections, we will analyze and explain this model from various perspectives using different methods. Due to the evaluation method used in our experiments, we have as many models as tested years. To simplify the study, we focus on the most recent year, which represents the most challenging test set and highlights the model's robustness.

In this section, we examine the selected inputs in two dimensions: features and lags. This selection is important to understand the which features and lags are considered to the problem, based on the model optimization criteria. For this purpose, we created a matrix in which the x-axis represents the lags and the y-axis represents the features. Each cell is green for a selected lagged feature, and blue otherwise. Finally, the features are divided into different groups to improve visual clarity.

4.4.1. Meteorological features

The meteorological input features are the first group studied. These features are divided into three categories: ground-level, ground-level lagged, and satellite represented in Figs. 2, 3, and 4, respectively.

It is interesting to note that, despite representing the same features, there are differences between the selected non-lagged and lagged features. This can be explained by the strong correlation between these features, which creates redundancy. As a result, the TSL chose only one version of the features or reduced the amount of information into one as much as possible.

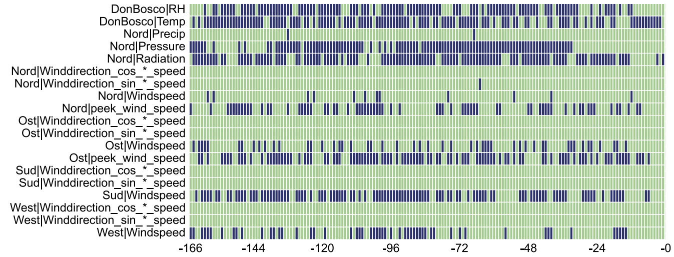

Fig. 2. Selected features by the TSL for the ground-level meteorological features.

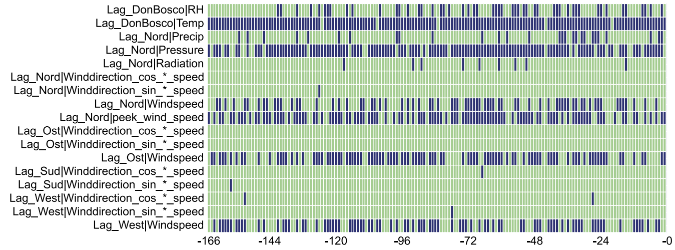

Fig. 3. Selected features by the TSL for the lagged ground level meteorological features.

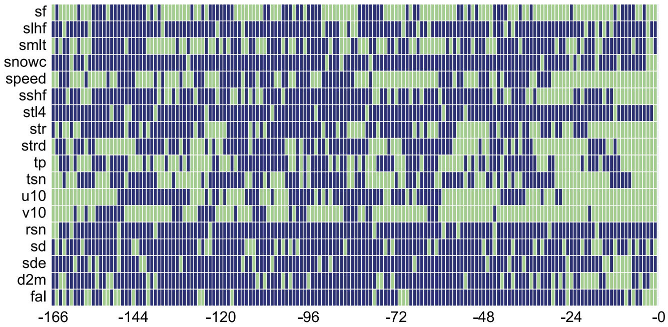

Fig. 4. Selected features by the TSL for satellite meteorological features.

For example, Fig. 2 shows that less than half of the lags for Don Bosco relative humidity and Nord radiation were selected, while most of the lags for the lagged versions of these features were chosen. Conversely, for West wind speed, the non-lagged feature was selected more frequently than its lagged counterpart.

However, in most cases, both groups of features are clearly aligned when selecting and removing certain features. The most frequently removed feature detected by the TSL in both groups is the Don Bosco temperature. In contrast, most features show significant overlap between the selected lags, except for the Nord pressure, where the most recent lags are more frequently selected in Fig. 2.

Finally, Fig. 4 shows a more sparse selection of data, indicating that the model considered most of the features irrelevant. This pattern may be due to spurious correlations between relevant features or the model getting stuck in a local optimum.

The most frequently selected feature appears to be the v10 component, which was selected in many of its lagged versions, similar to previous feature groups. Another significant feature is the u10 component, which shows a pattern in both recent and distant lags, possibly indicating a seasonal trend. Additionally, some features are only considered in the most recent lags. For instance, this is observed with wind speed, sshf, and str.

In conclusion, the first two groups of features are selected on almost all their lags excepting some features. In contrast, the final group shows a sparser selection, with only one or two key more frequent features. The following sections will further support this observation using feature importance techniques.

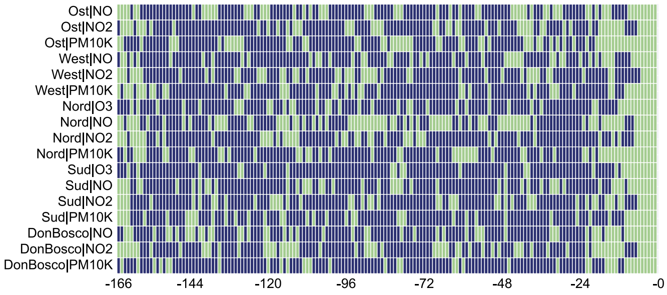

Fig. 5. Selected features by the TSL for the target features.

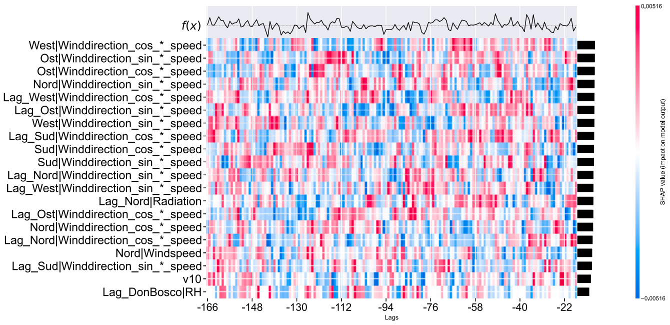

Fig. 6. SHAP values for top ten features by lag for NO pollutant.

4.4.2. *Target features*

The final group of features to analyze are the target variables we aim to predict. This group is especially important for understanding the model's performance.

In general, Fig. 5 does not show any particular feature being favored over others, which is expected since all features are used in the loss function. It is clear that the most recent lags, ranging from the last 6 to the last 24, are consistently selected. This pattern is common in time series forecasting, as recent lags are usually highly correlated with the predictions. A similar, though less pronounced, pattern is observed in the last lags, where most features are selected for periods of two to eight hours.

Although selection in non-extreme lags is sparse, there is a noticeable locality effect for some features during certain periods. This means that some lags are selected in groups. For example, the Nord NO pollutant shows this effect during several periods, particularly in the range of 96 to 48 hours before the cutoff.

In conclusion, this group of features is predominantly represented in recent time periods, with some exhibiting a locality effect at specific intervals.

4.5. *Shapley values*

To support the conclusions made in Section 4.4, this section quantifies the impact of different features on the forecast. We calculated the Shapley value for each input feature across each target value. The importance of each input feature was then averaged over all targets. Finally, we identified the top 20 features and represented their importance by lag in a heatmap.

Fig. 6 displays a heatmap of Shapley values. In this heatmap, the y-axis represents the features, and the x-axis represents the different lags. The cell values indicate the Shapley values, where higher values denote a positive influence on the targets, while lower values signify a negative influence.

The most important features are primarily meteorological, as discussed earlier. Specifically, lagged and non-lagged wind direction consistently ranks high in importance across all areas except for Don Bosco. Additionally, radiation and wind speed are crucial for the model in the Nord region. These meteorological features are expected to be significant because they influence pollutant generation and dispersion in the city.

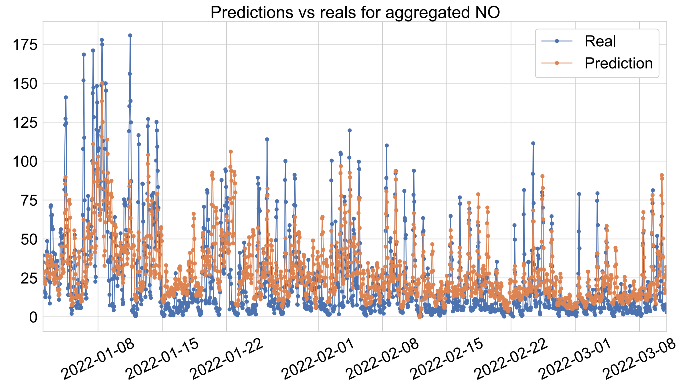

Fig. 7. Predictions vs reals NO.

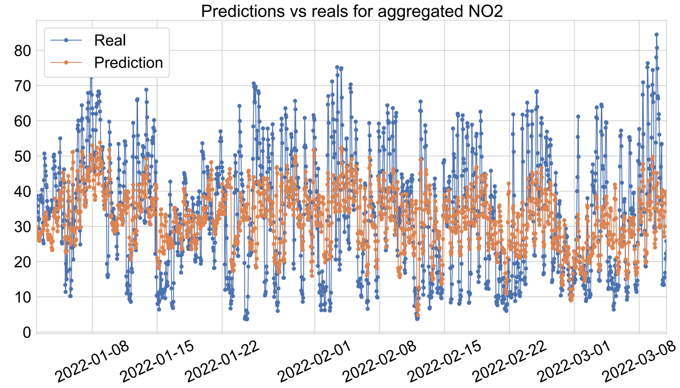

Fig. 8. Predictions vs reals NO_2.

While the same features are relevant across different areas, their importance relative to lags is inconsistent. There is no clear pattern of features showing consistent importance over specific lags. Among the top two features, some consecutive lags (around 60 to 70 and 100 to 120 hours ago) show a positive influence. This suggests that wind direction from noon to the end of the day may promote new pollutant formation.

Finally, despite filtering lags using TSL, the Shapley analysis indicates several lags in all features with almost zero importance. This issue arises from the method used and should be addressed in future work.

4.6. *Predictions*

This section presents the predictions made by the model. The predictions have been aggregated for each pollutant, regardless of the zone, to enhance visual clarity.

Fig. 7 shows the predicted and actual values for the NO pollutant. As discussed in Section 4.3, the NO pollutant exhibited the most variable results. This variability is evident in the figure, where periods of low pollution are followed by sudden extreme values. Notably, the first 12 days of January show a significant increase in NO levels, likely due to the effects of the holiday season. The model has captured this increase relatively well in its predictions.

It is important to note that NO levels remain quite low most of the time compared to other pollutants. As a result, the model performs worse in this case, as the time series lacks stationary patterns and behaves more chaotically.

Fig. 8 illustrates the behavior of the NO_2 pollutant. Unlike NO, NO_2 has fewer values near zero, with most of its levels hovering around 35 $\mu g/m^3$.

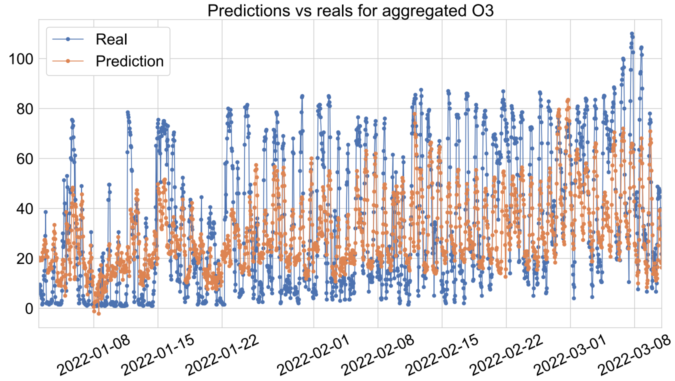

Fig. 9. Predictions vs reals O_3.

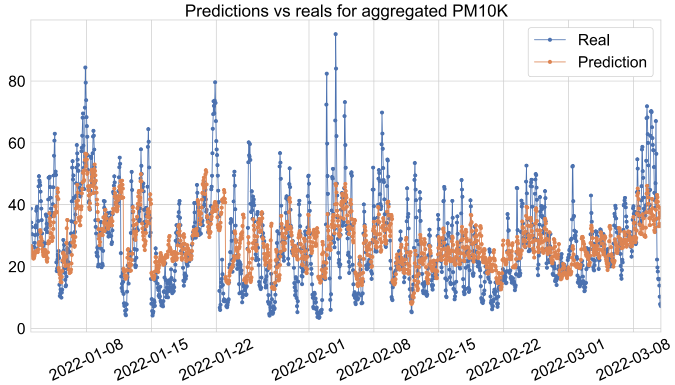

Fig. 10. Predictions vs reals PM10K.

In this case, the model predictions are more conservative, staying close to the mean with low variability. This results in better overall accuracy but misses extreme values, which could be critical for decision-making in some contexts.

Fig. 9 depicts the predictions for O_3. This pollutant shows a slight upward trend, possibly due to rising temperatures. Additionally, the data exhibits a strong daily seasonal pattern, which the model seems to have captured well. However, the model tends to avoid predicting the extreme high and low values.

Fig. 10 compares the predictions and actual values for the $PM_{10}K$ pollutant. This data shows a subtle weekly seasonal pattern, with higher values during weekends, which the model successfully captures. The dataset contains almost no values near zero, with a mean concentration of approximately 30 μg/m^3. In this case, the model captures more of the minimum values compared to the maximum ones, similar to its performance with NO_2 and O_3.

In conclusion, the model successfully learned the patterns of four different pollutants, accounting for seasonal trends, mean values, holiday effects, and more. Although there are some shortcomings, the results demonstrate the model's notable efficacy and flexibility.

5. Conclusions and future works

This paper analyzes the impact of a comprehensive set of features on air pollutant concentrations, using machine learning and explainability techniques. Specifically, a deep learning approach, with embedded feature selection using the TSL layer, delivered the best performance.

The results from the top-performing model underscore the critical role of meteorological features in explaining most past events, surpassing the importance of other input features. Among these, wind direction, wind speed, solar radiation, and relative humidity were particularly significant. Additionally, their lagged versions held similar importance, dominating the relevance rankings determined by the SHAP technique. Interestingly, endogenous features were less influential than meteorological ones, with the most recent events being the most frequently selected.

Future work could benefit from feature selection tailored to each output, helping to identify which inputs are most relevant for specific outputs. This is especially crucial in time series forecasting problems with multiple input/output features and future horizon predictions. Furthermore, improving both the quality and quantity of meteorological data seems to be the key to enhancing model performance. Incorporating additional ground-level meteorological information and integrating it with satellite data could potentially boost the model's efficacy.

CRediT authorship contribution statement

Manuel J. Jiménez-Navarro: Writing – original draft, Visualization, Software, Methodology, Formal analysis, Conceptualization. **Mario Lovrić:** Writing – review & editing, Supervision, Investigation, Data curation. **Simonas Kecorius:** Writing – review & editing, Supervision, Resources, Project administration, Formal analysis. **Emmanuel Karlo Nyarko:** Writing – review & editing, Writing – original draft, Visualization, Validation, Investigation, Data curation, Conceptualization. **María Martínez-Ballesteros:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Investigation, Formal analysis, Conceptualization.

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.

Acknowledgements

This research has been supported by the grant PID2020-117954RB-C22, PID2023-146037OB-C21 funded by MICIU/AEI/10.13039/501100011033. This work has also been supported by TED2021-131311B-C21 funded by MICIU/AEI/10.13039/501100011033 and by the MICIN/AEI/10.13039/501100011033/European Union NextGenerationEU/PRTR. This work was supported also by Horizon Europe (EDIAQI project #101057497).

Data availability

Data will be made available on request.

References

[1] C. Liu, R. Chen, F. Sera, A.M. Vicedo-Cabrera, Y. Guo, S. Tong, M.S.Z.S. Coelho, P.H.N. Saldiva, E. Lavigne, P. Matus, N.V. Ortega, S.O. Garcia, M. Pascal, M. Stafoggia, M. Scortichini, M. Hashizume, Y. Honda, M. Hurtado-Díaz, J. Cruz, B. Nunes, J.P. Teixeira, H. Kim, A. Tobias, C. Íñiguez, B. Forsberg, C. Åström, M.S. Ragettli, Y.-L. Guo, B.-Y. Chen, M.L. Bell, C.Y. Wright, N. Scovronick, R.M. Garland, A. Milojevic, J. Kyselý, A. Urban, H. Orru, E. Indermitte, J.J.K. Jaakkola, N.R.I. Ryti, K. Katsouyanni, A. Analitis, A. Zanobetti, J. Schwartz, J. Chen, T. Wu, A. Cohen, A. Gasparrini, H. Kan, Ambient particulate air pollution and daily mortality in 652 cities, N. Engl. J. Med. 381 (2019) 705–715, https://doi.org/10.1056/NEJMoa1817364.

[2] G.M. Lovett, T.H. Tear, D.C. Evers, S.E. Findlay, B.J. Cosby, J.K. Dunscomb, C.T. Driscoll, K.C. Weathers, Effects of air pollution on ecosystems and biological diversity in the eastern United States, Ann. N.Y. Acad. Sci. 1162 (2009) 99–135, https://doi.org/10.1111/j.1749-6632.2009.04153.x.

[3] A.M. Fiore, V. Naik, D.V. Spracklen, A. Steiner, N. Unger, M. Prather, D. Bergmann, P.J. Cameron-Smith, I. Cionni, W.J. Collins, S. Dalsøren, V. Eyring, G.A. Folberth, P. Ginoux, L.W. Horowitz, B. Josse, J.-F. Lamarque, I.A. MacKenzie, T. Nagashima, F.M. O'Connor, M. Righi, S.T. Rumbold, D.T. Shindell, R.B. Skeie, K. Sudo, S. Szopa, T. Takemura, G. Zeng, Global air quality and climate, Chem. Soc. Rev. 41 (2012) 6663–6683, https://doi.org/10.1039/C2CS35095E.

[4] N.L. Gilbert, S. Woodhouse, D.M. Stieb, J.R. Brook, Ambient nitrogen dioxide and distance from a major highway, Sci. Total Environ. 312 (2003) 43–46, https://doi.org/10.1016/S0048-9697(03)00228-6.

[5] L.J. Clapp, M.E. Jenkin, Analysis of the relationship between ambient levels of o3, no2 and no as a function of nox in the uk, Atmos. Environ. 35 (2001) 6391–6405, https://doi.org/10.1016/S1352-2310(01)00378-8.

[6] T. Wang, L. Xue, P. Brimblecombe, Y.F. Lam, L. Li, L. Zhang, Ozone pollution in China: a review of concentrations, meteorological influences, chemical precursors, and effects, Sci. Total Environ. 575 (2017) 1582–1596, https://doi.org/10.1016/j.scitotenv.2016.10.081.

[7] A. Churg, M. Brauer, Human lung parenchyma retains PM2.5, Am. J. Respir. Crit. Care Med. 155 (1997) 2109–2111, https://doi.org/10.1164/ajrccm.155.6.9196123.

[8] X. Gao, M. Jiang, N. Huang, X. Guo, T. Huang, Long-term air pollution, genetic susceptibility, and the risk of depression and anxiety: a prospective study in the UK Biobank Cohort, Environ. Health Perspect. 131 (2023) 1–10, https://doi.org/10.1289/EHP10391.

[9] S. Kumari, M. Jain, S.P. Elumalai, Assessment of pollution and health risks of heavy metals in particulate matter and road dust along the road network of Dhanbad, India, J. Health Pollut. 11 (2021) 210305, https://doi.org/10.5696/2156-9614-11.29.210305.

[10] W.H.O. (WHO), Healthy Environments for Healthier Populations: Why do They Matter, and What Can We Do?, World Health Organization, Geneva, 2019.

[11] M. Lovrić, M. Antunović, I. Šunić, M. Vuković, S. Kecorius, M. Kröll, I. Bešlić, R. Godec, G. Pehnec, B.C. Geiger, S.K. Grange, I. Šimić, Machine learning and meteorological normalization for assessment of particulate matter changes during the COVID-19 lockdown in Zagreb, Croatia, Int. J. Environ. Res. Public Health 19 (2022) 6937, https://doi.org/10.3390/ijerph19116937.

[12] I. Šimić, M. Lovrić, R. Godec, M. Kröll, I. Bešlić, Applying machine learning methods to better understand, model and estimate mass concentrations of traffic-related pollutants at a typical street canyon, Environ. Pollut. 263 (2020) 114587, https://doi.org/10.1016/j.envpol.2020.114587.

[13] Y. Han, J. Lam, V. Li, Q. Zhang, A domain-specific Bayesian deep-learning approach for air pollution forecast, IEEE Trans. Big Data (2022), https://doi.org/10.1109/tbdata.2020.3005368.

[14] W. Saeed, C. Omlin, Explainable AI (XAI): a systematic meta-survey of current challenges and future opportunities, Knowl.-Based Syst. 263 (2023) 110273, https://doi.org/10.1016/j.knosys.2023.110273.

[15] M. Jiménez-Navarro, M. Martínez-Ballesteros, F. Martínez-Álvarez, G. Asencio-Cortés, Explaining deep learning models for ozone pollution prediction via embedded feature selection, Appl. Soft Comput. (2024) 111504, https://doi.org/10.1016/j.asoc.2024.111504.

[16] N/A, Austrian Government Data, https://www.umwelt.steiermark.at/cms/ziel/2060750/DE/, 2023, (Accessed on 2023).

[17] S. Lundberg, S. Lee, A unified approach to interpreting model predictions, in: Proceedings of the International Conference on Neural Information Processing Systems, vol. 30, 2017, pp. 4765–4774.

[18] L.S. Shapley, A value for n-person games, in: H.W. Kuhn, A.W. Tucker (Eds.), Contributions to the Theory of Games, in: Annals of Mathematics Studies, vol. 2, Princeton University Press, 1953, pp. 307–317.

[19] A.R. Troncoso-García, I.S. Brito, A. Troncoso, F. Martínez-Álvarez, Explainable hybrid deep learning and coronavirus optimization algorithm for improving evapotranspiration forecasting, Comput. Electron. Agric. 215 (2023) 108387, https://doi.org/10.1016/j.compag.2023.108387.

[20] D. Gaspar, P. Silva, C. Silva, Explainable AI for intrusion detection systems: LIME and SHAP applicability on multi-layer perceptron, IEEE Access 12 (2024) 30164–30175, https://doi.org/10.1109/ACCESS.2024.3368377.

[21] M. Lovrić, K. Pavlović, M. Vuković, S.K. Grange, M. Haberl, R. Kern, Understanding the true effects of the COVID-19 lockdown on air pollution by means of machine learning, Environ. Pollut. 274 (2021) 115900, https://doi.org/10.1016/j.envpol.2020.115900.

[22] I. Gudelj, M. Lovrić, E.K. Nyarko, Modelling the daily concentration of airborne particles using 1D convolutional neural networks, Eng. Proc. 68 (2024), https://doi.org/10.3390/engproc2024068016.

[23] A.R. Troncoso-García, M.J. Jiménez-Navarro, F. Martínez-Álvarez, A. Troncoso, Ground-level ozone forecasting using explainable machine learning, in: Advances in Artificial Intelligence, Springer Nature Switzerland, 2024, pp. 71–80.

[24] C.O. Retzlaff, A. Angerschmid, A. Saranti, D. Schneeberger, R. Röttger, H. Müller, A. Holzinger, Post-hoc vs ante-hoc explanations: xai design guidelines for data scientists, Cogn. Syst. Res. 86 (2024) 101243, https://doi.org/10.1016/j.cogsys.2024.101243.

[25] E.M. Tibebe, P. Dondio, L. Longo, Explaining deep learning time series classification models using a decision tree-based post-hoc xai method, in: Proceedings of the xAI-2023 Late-Breaking Work, Demos and Doctoral Consortium Co-Located with the 1st World Conference on eXplainable Artificial Intelligence (xAI-2023), 2023, pp. 71–76.

[26] I. Kumar, B.K. Tripathi, A. Singh, Attention-based lstm network-assisted time series forecasting models for petroleum production, Eng. Appl. Artif. Intell. 123 (2023) 106440, https://doi.org/10.1016/j.engappai.2023.106440.

[27] K. Seo, J. Yang, Exploring candlesticks and multi-time windows for forecasting stock-index movements, in: Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing, SAC '23, Association for Computing Machinery, 2023, pp. 1100–1109.

[28] M. Wu, M.C.H.S. Parbhoo, M. Zazzi, V. Roth, F. Doshi-Velez, Beyond sparsity: tree regularization of deep models for interpretability, in: Proceedings of the 32nd AAAI Conference on Artificial Intelligence and 30th Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI Press, 2018.

[29] S. Rajab, V. Sharma, An interpretable neuro-fuzzy approach to stock price forecasting, Soft Comput. 23 (2019) 921–936, https://doi.org/10.1007/s00500-017-2800-7.

[30] B.N. Oreshkin, D. Carpov, N. Chapados, Y. Bengio, N-BEATS: neural basis expansion analysis for interpretable time series forecasting, in: Proceedings of the 8th International Conference on Learning Representations (ICLR), 2020, pp. 1–31.

[31] C. van Zyl, X. Ye, R. Naidoo, Harnessing explainable artificial intelligence for feature selection in time series energy forecasting: a comparative analysis of grad-cam and shap, Appl. Energy 353 (2024) 122079, https://doi.org/10.1016/j.apenergy.2023.122079.

[32] C. Pandey, A. Ji, T. Nandakumar, R.A. Angryk, B. Aydin, Exploring deep learning for full-disk solar flare prediction with empirical insights from guided grad-cam explanations, in: Proceedings of IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA), 2023, pp. 1–10.

[33] T.B. Çelik, O. İcan, E. Bulut, Extending machine learning prediction capabilities by explainable ai in financial time series prediction, Appl. Soft Comput. 132 (2023) 109876, https://doi.org/10.1016/j.asoc.2022.109876.

[34] A. Troncoso-García, M. Martínez-Ballesteros, F. Martínez-Álvarez, A. Troncoso, A new approach based on association rules to add explainability to time series forecasting models, Inf. Fusion 94 (2023) 169–180, https://doi.org/10.1016/j.inffus.2023.01.021.

[35] J. Zacharias, M. von Zahn, J. Chen, O. Hinz, Designing a feature selection method based on explainable artificial intelligence, EM 32 (2022) 2159–2184, https://doi.org/10.1007/s12525-022-00608-1.

[36] P. Dhal, C. Azad, A comprehensive survey on feature selection in the various fields of machine learning, Appl. Intell. 52 (2022) 4543–4581, https://doi.org/10.1007/s10489-021-02550-9.

[37] M.J. Jiménez-Navarro, M. Martínez-Ballesteros, F. Martínez-Álvarez, G. Asencio-Cortés, A new deep learning architecture with inductive bias balance for transformer oil temperature forecasting, J. Big Data 10 (2023) 80, https://doi.org/10.1186/s40537-023-00745-0.

[38] M.J. Jiménez-Navarro, M. Martínez-Ballesteros, I.S. Brito, F. Martínez-Álvarez, G. Asencio-Cortés, Embedded feature selection for neural networks via learnable drop layer, Log. J. IGPL (2024) jzae062, https://doi.org/10.1093/jigpal/jzae062.

[39] M.L. Linares-Barrera, M.J. Jimenez-Navarro, I.S. Brito, J.C. Riquelme, M. Martínez-Ballesteros, Evolutionary feature selection for time-series forecasting, in: Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing, SAC '24, Association for Computing Machinery, New York, NY, USA, 2024, pp. 395–399.

[40] M.J. Jiménez-Navarro, M. Martínez-Ballesteros, F. Martínez-Álvarez, G. Asencio-Cortés, Embedded temporal feature selection for time series forecasting using deep learning, in: Proceedings of International Work-Conference on Artificial Neural Networks, 2023, pp. 15–26.

[41] M. Lovrić, V. Petrić, K. Pavlović, A. Schopper, M. Vuckovic, Hourly Air Pollution Data for Graz, Austria, https://doi.org/10.5281/zenodo.7959116, 2023.

[42] H. Hersbach, B. Bell, P. Berrisford, G. Biavati, A. Horányi, J. Muñoz Sabater, J. Nicolas, C. Peubey, R. Radu, I. Rozum, D. Schepers, A. Simmons, C. Soci, D. Dee, J.-N. Thépaut, ERA5 hourly data on single levels from 1940 to present, https://doi.org/10.24381/cds.adbb2d47, 2019.

[43] J. Muñoz, Sabater, ERA5-Land hourly data from 1950 to present, https://doi.org/10.24381/cds.e2161bac, 2019, (Accessed on 2023).

[44] J.M. Sabater, Era5-land hourly data from 1950 to present, https://doi.org/10.24381/cds.e2161bac, 2019.

[45] V. Petrić, H. Hussain, K. Časni, M. Vuckovic, A. Schopper, v. Ujević Andrijić, S. Kecorius, L. Madueno, R. Kern, M. Lovrić, Ensemble machine learning, deep learning, and time series forecasting: improving prediction accuracy for hourly ambient for ambient air pollutants, Aerosol Air Qual. Res. 24 (2024) 230317, https://doi.org/10.4209/aaqr.230317.

[46] V.G. Costa, C.E. Pedreira, Recent advances in decision trees: an updated survey, Artif. Intell. Rev. 56 (2023) 4765–4800, https://doi.org/10.1016/j.patrec.2019.09.001.

[47] M. Xia, H.H. Cai, The driving factors of corporate carbon emissions: an application of the lasso model with survey data, Environ. Sci. Pollut. Res. Int. 30 (2023) 56484–56512, https://doi.org/10.1007/s11356-023-26081-7.

[48] A.X. Wang, S.S. Chukova, B.P. Nguyen, Ensemble k-nearest neighbors based on centroid displacement, Inf. Sci. 629 (2023) 313–323, https://doi.org/10.1016/j.ins.2023.02.004.

[49] P. Josso, A. Hall, C. Williams, T.L. Bas, P. Lusty, B. Murton, Application of random-forest machine learning algorithm for mineral predictive mapping of fe-mn crusts in the world ocean, Ore Geol. Rev. (2023) 105671, https://doi.org/10.1016/j.oregeorev.2023.105671.

[50] M. Niazkar, A. Menapace, B. Brentan, R. Piraei, D. Jimenez, P. Dhawan, M. Righetti, Applications of xgboost in water resources engineering: a systematic literature review, Environ. Model. Softw. (2024) 105971, https://doi.org/10.1016/j.envsoft.2024.105971.

[51] D. Narmandakh, C. Butscher, F.D. Ardejani, H. Yang, T. Nagel, R. Taherdangkoo, The use of feed-forward and cascade-forward neural networks to determine swelling potential of clayey soils, Comput. Geotech. 157 (2023) 105319, https://doi.org/10.1016/j.compgeo.2023.105319.

[52] S. Hochreiter, The vanishing gradient problem during learning recurrent neural nets and problem solutions, Int. J. Uncertain. Fuzziness Knowl.-Based Syst. 6 (1998) 107–116, https://doi.org/10.1142/S0218488598000094.

[53] P. Lara-Benítez, M. Carranza-García, J.C. Riquelme, An experimental review on deep learning for time series forecasting, Int. J. Neural Syst. 31 (2021) 2130001, https://doi.org/10.1142/S0129065721300011.

[54] J.F. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, A. Troncoso, Deep learning for time series forecasting: a survey, Big Data 9 (2021) 3–21, https://doi.org/10.1089/big.2020.0159.

[55] E.T. Habtemariam, K. Kekeba, M. Martínez-Ballesteros, F. Martínez-Álvarez, A Bayesian optimization-based LSTM model for wind power forecasting in the Adama District, Ethiopia, Energies 16 (2023) 2317, https://doi.org/10.3390/en16052317.

[56] J. Snoek, H. Larochelle, R.P. Adams, Practical Bayesian optimization of machine learning algorithms, in: Proceedings of the Advances in Neural Information Processing Systems, 2012, pp. 1723–1731.