Public AI Challenge: ChInquinArIA?

From Explainability to High-Resolution Forecasting: A Dual-Approach Framework for Air Quality Prediction

Alberto Catalano	Alessandro Delle Site	Elisa Negrini	Ettore Miglioranza	Federico Rubbi
alberto.catalano@studenti.unitn.it	alessandro.dellesite@studenti.unitn.it	elisa.negrini@studenti.unitn.it	ettore.miglioranza@studenti.unitn.it	federico.rubbi@studenti.unitn.it

Nicolò Cecchin	Stefano Genetti	Elena Tomasi	Elisa Malloci	Massimo Cassiani
nicolo.cecchin@studenti.unitn.it	stefano.genetti@unitn.it	eltomasi@fbk.eu	elisa.malloci@provincia.tn.it	massimo.cassiani@unitn.it

📝**Miro Board:** https://miro.com/app/board/uXjVJGHwMO8=/

🤖**GitHub Repository:** https://github.com/StefanoGenettiUniTN/appa-chinquinaria.git

1. Introduction

Air quality results from a complex interplay of multiple factors, including local emission sources (transportation, heating systems, industrial activities), atmospheric chemical and physical transformations, meteorological conditions, and long-range pollutant transport (Karagulian et al., 2015; Monks et al., 2009). The outcome of this complexity ultimately determines the concentration, composition, and persistence of airborne substances that people breathe and that affect the entire ecosystem (Lelieveld et al., 2015; Fowler et al., 2009). In Trentino, air quality monitoring is conducted through a network of eight fixed stations managed by the Provincial Environmental Protection Agency (APPA). Long-term data from these stations have clearly shown recurring episodes of pollutant transport, particularly of fine particulate matter (PM_{10}) from the Po Valley, driven by specific meteorological configurations. However, these phenomena have so far been assessed only qualitatively, without a robust quantification of their frequency or contribution to local pollution levels. The recent EU Air Quality Directive, which significantly lowers pollutant concentration limits by 2030, poses new challenges for the region and calls for updated management strategies based on a more accurate understanding of pollutant transport phenomena.

The objective of this work is to apply Artificial Intelligence (AI) methods to support APPA in studying these phenomena more systematically. In particular, we focus on identifying relevant drivers of PM_{10} concentrations and producing reliable forecasts of future PM_{10} levels. Forecasting is essential for assessing how often and for how long regulatory thresholds may

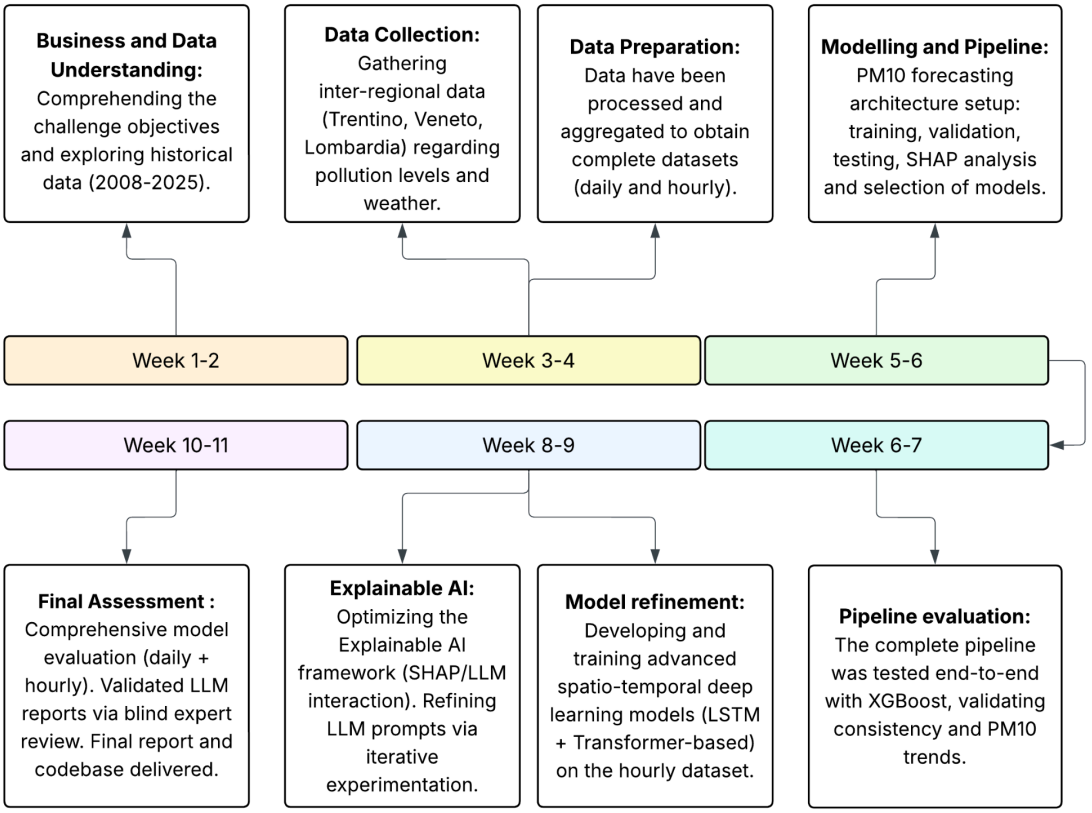
be exceeded, enabling the agency to take proactive measures. Yet many existing forecasting approaches rely on black-box models, whose internal mechanisms are difficult for end users to interpret (Reichstein et al., 2019). APPA experts, however, require not only accurate predictions but also clear explanations of the physical and environmental factors underlying them. To meet this dual need, we construct two comprehensive datasets of air quality and meteorological variables across Northern Italy (2013–2025): one aggregated daily and one aggregated hourly. The daily dataset is used to develop an innovative Explainable Artificial Intelligence (XAI) framework for understanding PM_{10} dynamics. Our methodology combines widely used predictive models to regress relevant features, SHAP for feature importance quantification (Lundberg & Lee, 2017), and Large Language Models (LLMs) for textual interpretation of local and large-scale meteorological configurations driving pollutant transport. By analyzing results over short temporal windows, the method captures local patterns that would otherwise be overlooked in broader analyses. In parallel, the hourly dataset is used to investigate more advanced recently introduced state-of-the-art deep-learning forecasting architectures, including LSTM-based and Transformer-based models, which provide more robust medium- and long-term predictions (Li et al., 2017; Liang et al., 2023). Though integrating explainability into these deep models remains an open research direction, our results highlight their strong predictive potential (Arrieta et al., 2020). The overall framework is evaluated using quantitative forecasting metrics as well as expert feedback from APPA. Both confirm the effectiveness, interpretability, and practical relevance of the proposed approach.

The remainder of this report is organized as follows:

- Section 2 provides a schematic overview of the activities carried out during the ten weeks of the challenge.
- Section 3 describes the data collection process and the characteristics of the resulting datasets.
- Section 4 presents the proposed methodology.
- Section 5 outlines the experimental setup used to evaluate the proposed approach, including the performance metrics, hyperparameters, and computational environment.
- Section 6 reports and analyzes the obtained results.
- Section 7 discusses the potential impact of the developed solution.
- Section 8 highlights the limitations of the current framework and suggests directions for future work.
- Finally, Section 9 concludes the report by summarizing the main findings.

2. Schematic overview of the 10-week project workflow

The project followed an iterative data-science process, with phases of business understanding, data comprehension, preparation, modelling, and evaluation. Each week represented a concrete progression within this structured process, ensuring a clear and progressive organization of the team's work over the ten weeks.



3. Dataset

To address this challenge and effectively analyze PM_{10} dynamics, it is necessary to integrate data from diverse geographic and meteorological sources. A key contribution of this work is the construction of two comprehensive final datasets: one with daily resolution (**dataset_day**) and another with hourly resolution (**dataset_hour**). These datasets were created by aggregating information from multiple environmental monitoring networks and meteorological data providers.

The daily dataset contains, for each APPA monitoring site and day, PM10 concentrations from APPA and neighboring European environmental agencies, local meteorological variables from Meteotrentino at the closest weather station, and large-scale meteorological descriptors from ERA5 (boundary layer height and pressure-level fields at 950, 850 and 550 hPa).

The hourly dataset covers 37 air-quality stations in Trentino, Lombardy, Veneto and South Tyrol and, for each (datetime, station_code), includes PM10 together with ERA5-Land near-surface meteorology (pressure, precipitation, solar radiation, 10 m wind, 2 m temperature), boundary layer height, and ERA5 pressure-level variables (temperature, humidity and wind components at 950, 850 and 550 hPa). These datasets were created by aggregating information from multiple environmental monitoring networks and meteorological data providers.

3.1. Agenzia Provinciale per la protezione dell'ambiente - APPA

The primary data source used in this study, serving as the ground-truth reference for evaluating the effectiveness of our approach in analyzing PM_{10} behavior, is the APPA historical pollutant concentration database. The data were obtained directly from the APPA open data portal and originate from a network of fixed monitoring stations equipped with

certified instruments for measuring atmospheric pollutants. Our analysis focuses exclusively on PM_{10} as the target pollutant. Additional pollutants, such as NO_2, are not considered, as they fall outside the scope of this study. The spatial domain comprises eight monitoring stations selected to represent the environmental variability of the province, covering urban, rural, industrial, and high-altitude settings.

- **Parco S. Chiara** (Urban Background - Trento)
- **Via Bolzano** (Urban Traffic - Trento)
- **Piana Rotaliana** (Rural background - Vineyard)
- **Rovereto** (Urban Background)
- **Borgo Valsugana** (Suburban Background)
- **Riva del Garda** (Suburban Background - Lakeside)
- **A22 (Avio)** (Rural Traffic - Highway)
- **Monte Gaza** (Rural background - High-altitude Background)

Figure 1: Example of APPA monitoring station.

3.2. Meteotrentino

Historical meteorological data from the Meteotrentino archive were used to characterize local atmospheric dynamics. The extracted feature set included ground-level temperature, relative humidity, precipitation, wind vector components (speed and direction), and solar radiation. Telemetry was available at both daily and hourly resolutions, and only datasets that had undergone formal validation and quality-control procedures by the provider were used. To integrate these spatially disjoint measurements with the APPA air-quality network, we applied a nearest-neighbor spatial alignment. For each APPA monitoring site, we computed the Haversine distance to every meteorological station and assigned the closest station to that site. This procedure yielded geolocated, station-specific feature vectors for every timestamp in both the daily and hourly datasets. Data completeness varied across stations and variables. Stations exceeding a predefined threshold of missing values were removed to preserve dataset integrity. For the remaining time series, missing entries were solved using temporal interpolation or forward-filling, depending on the nature of the variable. In the final data products, Meteotrentino measurements are used in the daily-resolution dataset to describe local weather at each APPA target station, whereas the hourly dataset relies on ERA5-based meteorological fields described in *Section 3.4*.

3.3. Regional and European Environmental Agencies

To characterize pollutant transport dynamics from neighbouring regions such as the Po Valley, we integrated external air-quality datasets into our analysis framework. For the daily-resolution dataset, we used the centralized services of the European Environment Agency (EEA), in particular the Up-to-date air quality data explorer to retrieve daily measurements. In contrast, obtaining hourly-resolution data required more granular records that were not consistently available through the central European repository. We therefore requested high-frequency pollutant measurements directly to relevant regional agencies: ARPA Veneto, ARPA Lombardia, and APPA Bolzano. Monitoring stations were selected according to two criteria: (i) minimizing the distance to the Trentino border to ensure representative boundary conditions, and (ii) prioritizing stations with the lowest proportion of missing data to guarantee robust time-series coverage. Acceptable missing-data thresholds were handled as described in Section 3.2. In practice, daily PM_{10} series from EEA and APPA air monitoring stations (*Figure 2*) contribute to the daily dataset, while high-frequency PM_{10} records from APPA, ARPA Veneto, ARPA Lombardia and the Bolzano Environment Agency are used to build the 37-station hourly dataset (*Figure 3*).

Figure 2: target APPA monitoring stations (red) and selected European Environmental Agency (EEA) stations (blue) used to capture external pollutant concentrations with daily resolution.

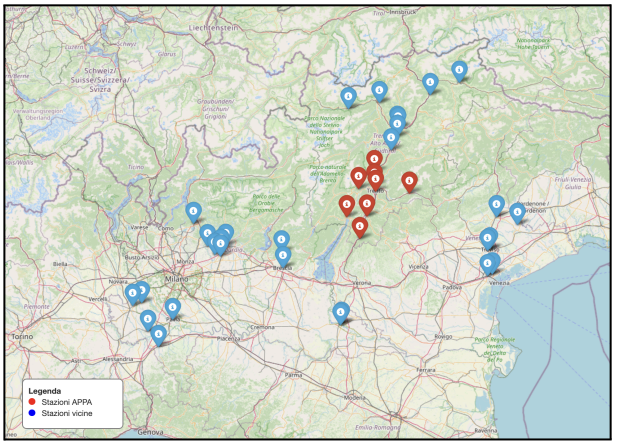


Figure 3: target APPA monitoring stations (red) and selected regional air quality monitoring stations (blue) with hourly resolution.

3.4. Copernicus ERA5 Reanalysis

To represent the large-scale synoptic forcing that drives pollutant transport, we incorporated meteorological fields from Copernicus ERA5, the fifth-generation global atmospheric reanalysis produced by the European Centre for Medium-Range Weather Forecasts (ECMWF). ERA5 is widely regarded as the state-of-the-art historical climate dataset, generated by collecting extensive global observations into an advanced numerical weather prediction system. The result is a spatially complete, temporally consistent four-dimensional reconstruction of atmospheric conditions, enabling characterization of meteorology even in regions lacking ground-based measurements. For this study, we extracted ERA5 variables over the 2008–2025 period, focusing on features that govern atmospheric stability and transport:

- Boundary Layer Height (BLH): used to quantify the vertical extent available for pollutant dispersion and to identify periods of suppressed mixing or inversion-driven stagnation.
- Atmospheric Profile: relative humidity, temperature, and wind components (zonal u and meridional v) were taken from three isobaric surfaces: 950 hPa (near-surface/valley layer), 850 hPa (lower troposphere), and 550 hPa (mid-troposphere), to capture different layers of the transport environment (Seidel et al., 2012).

3.4. Copernicus ERA5 Land

In addition to the pressure-level reanalysis fields, we used ERA5-Land to represent near-surface meteorological conditions at the coordinates of each air-quality station. ERA5-Land is a high-resolution land-surface reanalysis derived from the ERA5 system, providing spatially consistent, gap-free estimates of surface meteorology. For each station and timestamp in the hourly dataset, we sampled the grid cell closest to the monitoring site and extracted near-surface temperature, humidity and 10 m wind components (from which wind speed and direction are derived), mirroring the set of variables used at the 950, 850 and 550 hPa levels. Hourly measurements from ground-based weather stations were not used in this context, because their records contained substantial gaps and, in many cases, the stations were geographically distant from the corresponding air-quality sites.

This multi-level framework allows to model both near-surface stability dynamics and the synoptic-scale transport pathways that ground-based measurements often fail to detect. For daily dataset, ERA5 variables are sampled at daily resolution at the locations of the APPA monitoring sites, providing boundary layer height and pressure-level fields consistent with the PM_{10} and Meteotrentino records. For hourly dataset, ERA5-Land near-surface variables and boundary layer height are stored at hourly frequency for each air-quality station, while pressure-level fields originally available every three hours are linearly interpolated in time to obtain an hourly grid of temperature, humidity and wind components at 950, 850 and 550 hPa.

4. Methodology

The aim of this work is twofold. First, we develop and apply an XAI framework to better understand the dynamics of PM_{10} concentrations. Second, we investigate the effectiveness of recent deep learning architectures for robust PM_{10} forecasting. The discussion is therefore

structured into two parts. For the daily dataset, we design an innovative XAI methodology to identify and interpret the key features that drive PM_{10} behavior (*Section 4.1*). For the hourly dataset (which provides richer information due to its higher temporal resolution), we explore state-of-the-art deep learning models aimed at improving mid- long-term forecasting accuracy (*Section 4.2*). Integrating advanced XAI techniques with the employed deep learning architectures remains an active and promising research direction and represents a valuable opportunity for future extensions of this study.

4.1. Explainable Artificial Intelligence for understanding PM_{10} behavior

Figure 4 presents a schematic overview of the proposed framework.

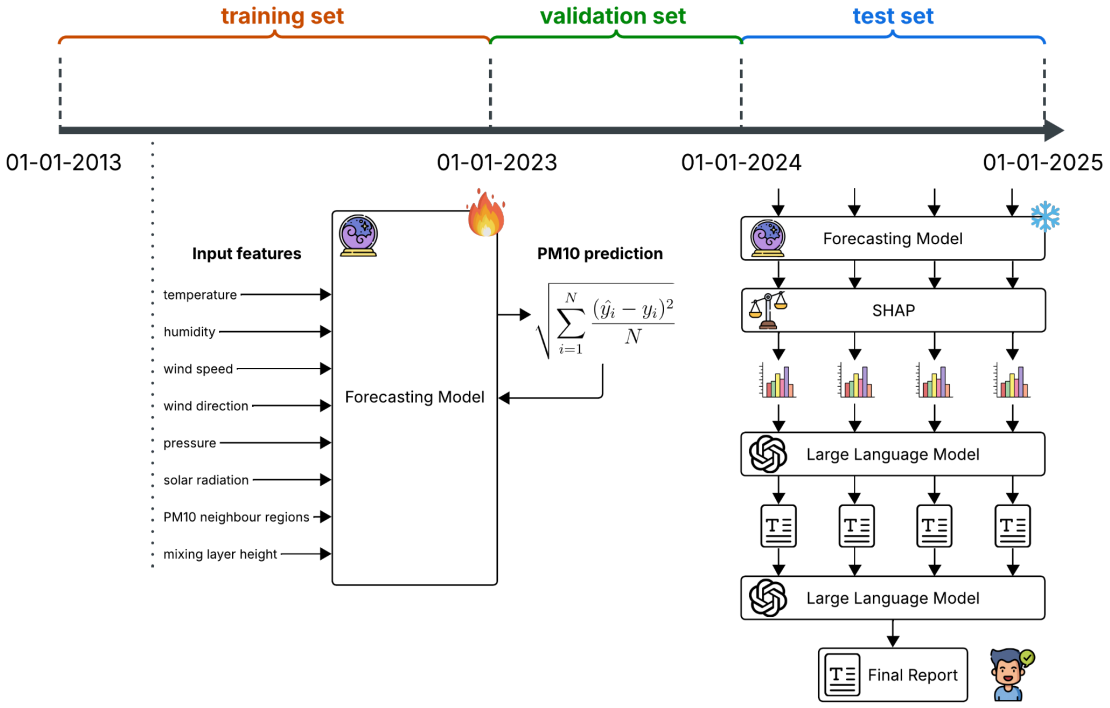


Figure 4: Conceptual scheme of the proposed methodology.

Using the daily dataset, our analysis covers the period from January 2013 to January 2025, during which the collected data are both complete and extensive enough to yield meaningful insights. The time frame is divided into three segments: a training set (from 01-01-2013 to 31-12-2022), a validation set (from 01-01-2023 to 31-12-2024), and a test set (from 01-01-2024 to 31-12-2024).
Within the training set, we train a predictive model M, which takes as input the feature set

$$X = \{f_1, f_2, \dots, f_K\}$$

identified as significant predictors of PM_{10} concentration Y (see *Section 3*), and produces a prediction of PM_{10} values over time:

$$M(f_1, f_2, \dots, f_K, t) = PM_{10}(t)$$

The predictive models evaluated in our framework are described in *Section 5.2*. The inference process is performed individually for each APPA monitoring station, allowing the

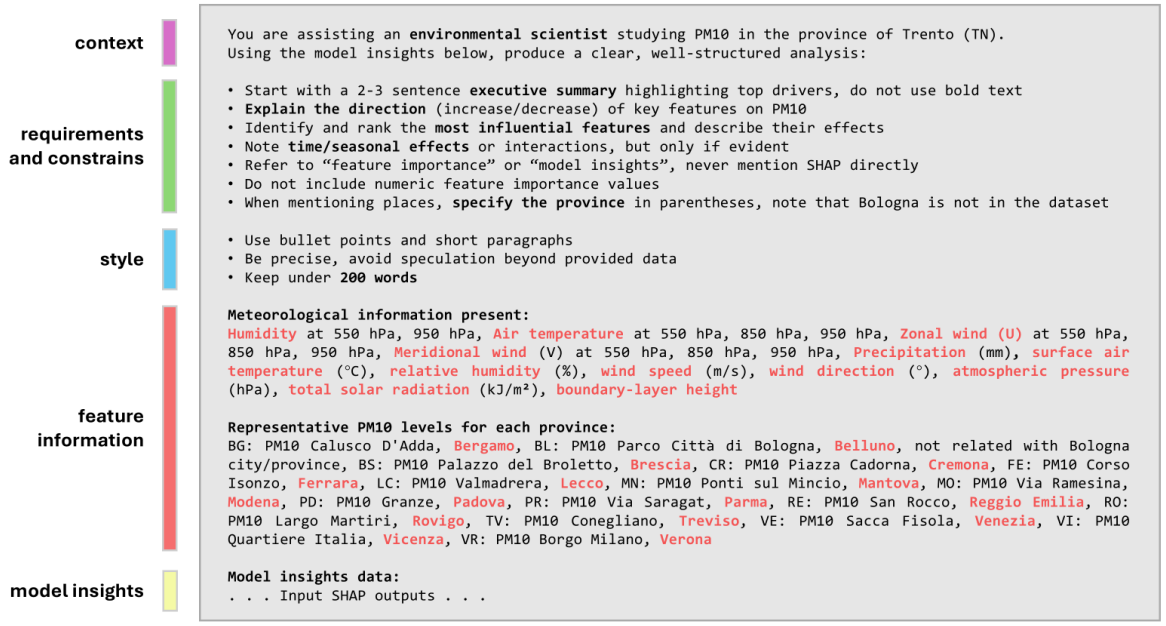
model to learn the underlying relationships between the features X and PM_{10} levels Y in Trentino.
After the training phase, the model M is applied to the test set. The primary goal of our methodology is to analyze the target period and generate a comprehensive textual report for the end user. To gain a deeper understanding of air pollutant dynamics, it is essential to focus on shorter time intervals. Indeed, analyzing overly long periods may obscure important local trends and variations. To capture these localized behaviors more effectively, we partition the test set into a series of time windows

$$W = \{w_1, w_2, \dots, w_N\}$$

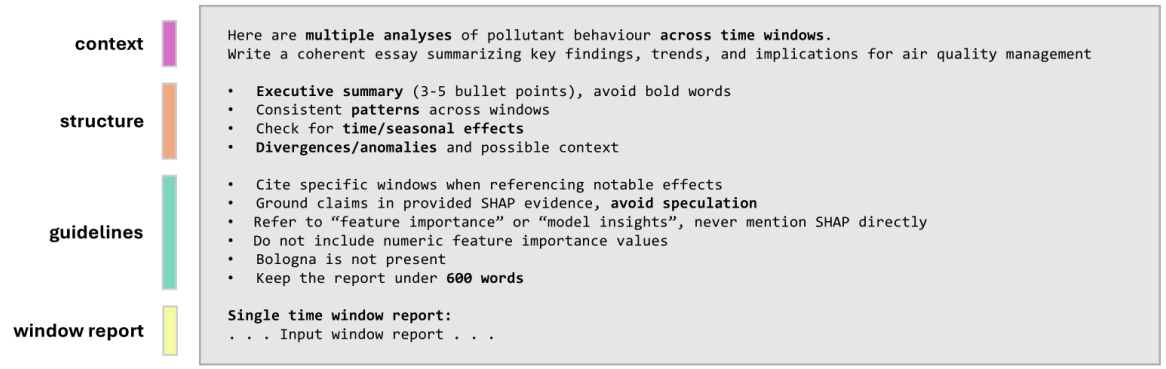
For each window w_j, the model M predicts PM_{10} levels based on the corresponding subset of input features. On top of these predictions, we apply SHapley Additive exPlanations (SHAP), a widely used XAI method, to quantify the contribution of each feature to the model's output. SHAP provides a feature importance report $S_1, S_2, \dots, S_N$ for each time window, where each report S_i consists of K numerical values (one per feature) representing its relative impact. The higher a SHAP value, the greater the feature's influence on the PM_{10} prediction. Although SHAP enhances model explainability, it can still be difficult for human operators to manually interpret a large number of SHAP reports comprehensively. This can limit both understanding and trust in the model's outputs. To overcome this limitation, we leverage a LLM to automatically generate textual explanations $d_1, d_2, \dots, d_N$ from the SHAP reports, providing intuitive and human-readable insights for each time window. Finally, to avoid overwhelming users with multiple potentially lengthy window-level reports, we employ the LLM once more to synthesize a final comprehensive summary D, which consolidates the key information extracted from all windows into a coherent analytical essay.

The interaction with the LLM is conducted exclusively through text, making prompt engineering crucial for obtaining accurate, coherent, and user-friendly explanations. Through iterative experimentation, we evaluated multiple prompt designs and refined them to optimize the quality of the generated outputs. The final versions of the prompts are structured as follows:

PROMPT 1: used to generate the textual descriptions $d_1, d_2, \dots, d_N$ for each time window.



PROMPT 2: used to generate the final aggregated report D.



As detailed in *Section 5.3*, in our experimental session, we tested multiple pretrained LLMs, both proprietary and open-source.

4.2. Deep learning for robust PM_{10} forecasting

For the hourly dataset, we explored advanced black-box deep learning architectures capable of capturing complex temporal dependencies and delivering accurate mid- to long-term forecasts. Specifically, we implemented Long Short-Term Memory (LSTM) networks and Transformer models, both of which are well-suited for time series prediction. LSTM networks were selected for their ability to model long-range sequential dependencies while mitigating the vanishing-gradient problem, making them particularly effective for capturing temporal dynamics over extended periods (Hochreiter et al., 1997). In parallel, Transformer-based architectures were employed to exploit self-attention mechanisms, enabling the models to efficiently learn temporal patterns and richer representations across long horizons without relying on recurrent operations (Vaswani et al., 2017). A more detailed description of these models is provided in *Section 5* (Experimental Setup).
The hourly dataset was split chronologically to ensure robust evaluation of predictive performance:

- Training set: 1 January 2014 - 31 December 2021
- Validation set: 1 January 2022 - 31 December 2022
- Test set: 1 January 2023 - 31 December 2024

The overall modeling workflow is schematized in *Figure 5*. Models were trained using a combination of lag features, autoregressive inputs, rolling windows, and embeddings, enabling the capture of non-linear patterns and the transformation of features into dense representations. All continuous variables were normalized using either MinMax or Standard scaling to ensure stable gradient behavior during training.

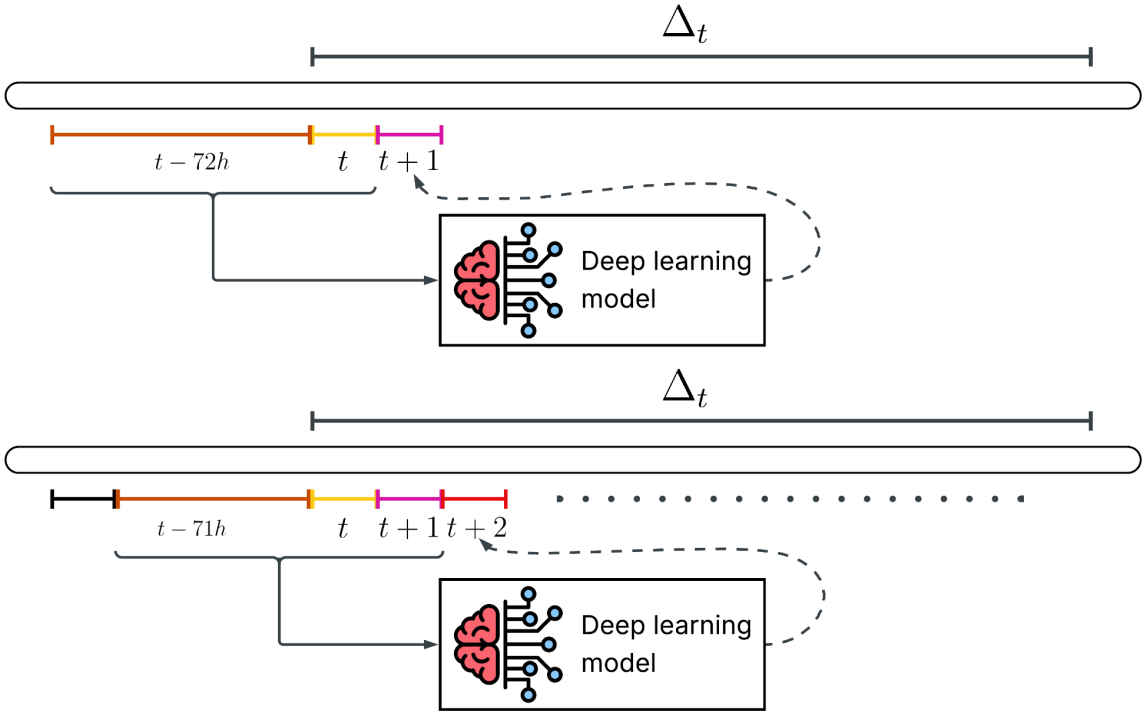


Figure 5: Deep learning models (LSTM and Transformer) use lagged features and autoregressive inputs to forecast pollutant levels over mid- to long-term horizons.

Let y_t denote the target variable at hour t. An autoregressive (AR) formulation can be expressed as:

$$y_t = f(y_{t-1}, y_{t-2}, \dots, y_{t-p}, X_t) + \epsilon_t$$

where p represents the number of lagged observations providing historical context, X_t includes exogenous features (e.g., wind speed and direction at 550hPa, PM10 concentration in neighbouring stations, etc.) at time t, $f(\cdot)$ is the model mapping past information to the current value, and ϵ_t is the error term.

In our setup, predictions at hour t are based on a sliding window of the previous 72 hours (from $t-72$ to $t-1$). For the first prediction, the model uses only observed values. After predicting yt, the window shifts forward by one hour, including the newly predicted value along with the preceding 71 observed values as autoregressive inputs. This iterative process continues to generate forecasts up to a desired horizon Δt.

$$\hat{y}_{t+k} = f(\hat{y}_{t+k-1}, \hat{y}_{t+k-2}, \dots, \hat{y}_{t+k-p}, X_{t+k}), \quad k = 1, 2, \dots, \Delta t$$

This autoregressive forecasting procedure allows the model to propagate predictions forward while integrating both historical observations and previously forecasted values, effectively capturing temporal dependencies over extended horizons. In our experiments, we evaluated multiple values of Δt to assess model performance across different forecasting horizons, as detailed in *Section 5*.

5. Experimental setup

This section describes the experimental setup used to assess the effectiveness of our proposed approach, including the machine learning models employed and the performance evaluation metrics.

5.1. Computational setup

We performed our experiments on Google Colab Pro+, which provides up to 52 GB of RAM, 2 vCPUs, and an NVIDIA V100 GPU. Most components of our project can be executed on a standard laptop; however, the open-source LLM requires substantial computational resources available only through a Colab Pro subscription. Our codebase is completely written in Python and is made publicly available on GitHub: https://github.com/StefanoGenettiUniTN/appa-chinquinaria.git.

5.2. Forecasting models

The implemented forecasting models can be broadly categorized into two families: classical machine learning models, which offer interpretable insights and are suitable for tabular data, and advanced deep learning architectures, which are designed to capture complex temporal dependencies in high-frequency data.

For the daily-aggregation dataset, we employed widely used predictive models to regress daily PM_{10} concentrations using the relevant day-level features. The goal was to capture the nonlinear dependencies that drive PM_{10} dynamics and to support interpretable insights within our XAI framework. Specifically, our experimental evaluation included the following models: `Random Forest` (scikit-learn library), `LightGBM` (LigthGBM library), `XGBoost` (XGBoost library), and a `Multi-Layer Perceptron (MLP)` (MLP pytorch). All model hyperparameters were optimized using `Optuna` (Akiba et al., 2019) for systematic hyperparameter tuning.

In contrast, for the hourly dataset, we evaluated a diverse set of Deep Learning architectures, ranging from established Recurrent Neural Networks (RNNs) to state-of-the-art Transformer-based models. For these models, we adopted the default hyperparameter configurations provided by their implementations. In particular, the following architectures were tested:

- Among RNNs, we tested `LSTM` (Hochreiter et al., 1997) and `GRU` (Cho et al., 2014), which process data sequentially and rely on internal gating mechanisms to manage memory over time. Their inductive bias strongly favors temporal order, making them effective for sequence modeling but often computationally constrained by their inability to parallelize processing and limited in capturing extremely long-range dependencies.
- In the Transformer landscape, we explored `PatchTST` (Nie et al., 2023), `iTransformer` (Liu et al., 2024), `Crossformer` (Zhou et al., 2024), `TimeXer` (Wang et al., 2024), and `Samformer` (Ilbert et al., 2024). These architectures leverage

self-attention mechanisms to model global dependencies without the constraints of sequential processing. In particular:

- `PatchTST` segments time series into sub-series "patches" (tokens), employing channel-independence to learn shared embeddings across variables. This allows it to capture local semantic patterns while reducing computational complexity.
- `iTransformer` inverts the standard embedding strategy by treating the entire time series of each variate as a single token, explicitly focusing the attention mechanism on modeling multivariate correlations rather than just temporal steps.
- `Crossformer` and `TimeXer` utilize specialized attention layers to capture both cross-dimension and cross-time dependencies, optimizing the representation of complex temporal dynamics.
- `Samformer` combines a shallow Transformer encoder with channel-wise self-attention and Sharpness-Aware Minimization (SAM) to improve both efficiency and generalization. Instead of attending over time steps, `Samformer` applies attention across feature dimensions, capturing cross-variable correlations with lower computational cost, and pairs this with reversible instance normalization and SAM-based optimization to encourage flatter minima and more robust long-horizon forecasts.

5.3. Large Language Models

Within the XAI framework, we evaluated both proprietary and open-source state-of-the-art LLMs. The proprietary model used is `OpenAI's GPT-4.1` (Kumar et al. 2025), while the open-source model is the Hugging Face implementation of `Mistral-7B-Instruct-v0.2` (Jiang et al., 2023). No fine-tuning or modifications were applied to the pre-trained models, and `GPT-4.1` was accessed via its public API.

5.4. Evaluation metrics

To assess the quality of the proposed methodology, the evaluation is twofold. First, we measure the performance of the forecasting model in predicting/regressing pollutant concentrations. Notably, the reliability of the feature-importance quantification depends directly on the accuracy of these predictions. Second, we assess the quality of the generated reports produced by the Large Language Models (LLMs).

Forecasting model performance. We evaluate the predictions using the following metrics:

- Mean Absolute Error (MAE): measures the average magnitude of errors in the predictions, without considering their direction.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \widehat{y}_i\right|$$

- Root Mean Squared Error (RMSE): measures the square root of the average squared differences between predicted and true values, penalizing larger errors more heavily.

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

- Dynamic Time Warping (DTW): A distance measure that compares two temporal sequences by allowing nonlinear alignments in time. DTW is especially useful for time series that may be misaligned or exhibit temporal shifts (Li et al., 2021).
- Execution time (in seconds): The total computational time required to train the forecasting model, reflecting the method's efficiency.

Large Language Model performance. Quantitative evaluation of LLM performance is not feasible in our use case, as the task involves subjective qualitative judgment. To compare `OpenAI's GPT-4.1` with `Mistral-7B-Instruct-v0.2`, we therefore rely on expert assessment. Specifically, we present a set of generated reports from both LLMs without revealing their source to seven people and ask them to indicate which report they prefer. This blind qualitative evaluation helps determine which model produces clearer, more accurate, and more useful reports according to expert judgment.

6. Results and discussion

In line with the twofold approach of our intervention for APPA, we structured our experimental evaluation into two subsections. In *Section 6.1*, we present and discuss the results obtained on the daily-aggregated dataset using the XAI framework. In *Section 6.2*, we present and analyze the results of the deep learning models applied to the hourly dataset.

6.1. Explainable Artificial Intelligence framework results on the daily dataset

Table 1 presents a quantitative comparison of the predictive algorithms applied to the training, validation, and test splits of the dataset aggregated at a daily resolution. Overall, all well-known predictive models demonstrate comparable performance. Notably, `Random Forest` exhibits a significantly longer training time compared to the others. `LightGBM`, on the other hand, achieves the best balance between training efficiency and predictive accuracy, offering very fast training while attaining the lowest MAE and RMSE on the training and validation sets, as well as the best DTW score on the validation set.

Split	Algorithm	MAE	RMSE	DTW	TRAINING TIME (SECONDS)
Training set (2013-2023)	RANDOM FOREST	**1.21**	**1.81**	**298.82**	209.84
	LIGHTGBM	2.88	4.00	594.45	**0.46**
	XGBOOST	2.71	3.75	563.94	13.60
	MLP	3.97	5.76	804.22	64.13
Validation set (2023-2024)	RANDOM FOREST	4.05	6.16	267.30	-
	LIGHTGBM	**3.93**	**5.87**	**250.50**	-
	XGBOOST	3.94	5.90	255.76	-
	MLP	4.00	5.95	262.90	-
Test set (2024-2025)	RANDOM FOREST	3.97	5.88	253.39	-
	LIGHTGBM	**3.87**	**5.71**	245.36	-
	XGBOOST	3.88	5.78	**244.03**	-
	MLP	4.36	6.48	265.45	-

Table 1: Results on the daily-aggregated dataset of the predictive models regressing PM_{10} from relevant features using different algorithms. Best values for each metric in bold.

Figure 6 shows the predicted curves of each model across three selected test set windows: Window 2 (February 2024), Window 5 (May 2024), and Window 9 (September 2024). Consistent with the quantitative results reported in *Table 1*, all models closely follow the actual PM_{10} trends. Notably, the MLP model stands out for its ability to accurately capture the peak at the beginning of Window 2.

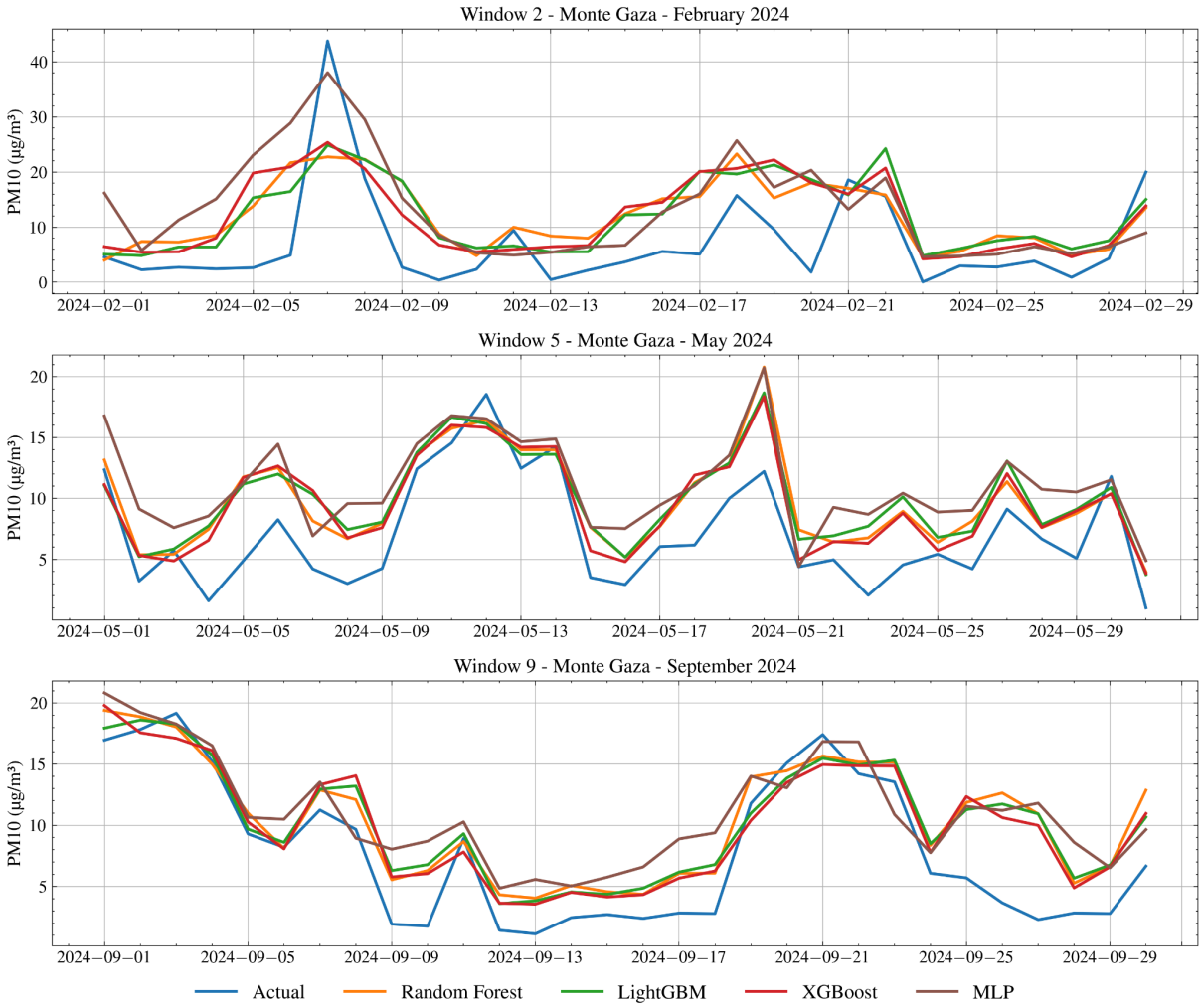


Figure 6: Performance of the regression algorithms on the test set at the Monte Gaza APPA PM_{10} monitoring station for daily-aggregated data. Results are shown for three windows: February 2024 (Window 2), May 2024 (Window 5), and September 2024 (Window 9). Each predictive model is represented by a distinct color curve, while the blue line indicates the observed PM_{10} concentrations.

For each of the twelve windows into which the test set was divided to capture local phenomena, we quantified the importance of each feature for the model's predictions using SHAP, producing a separate SHAP report for each window. *Figure 7* shows a representative example of SHAP feature importance for Window 1, corresponding to January 2024.

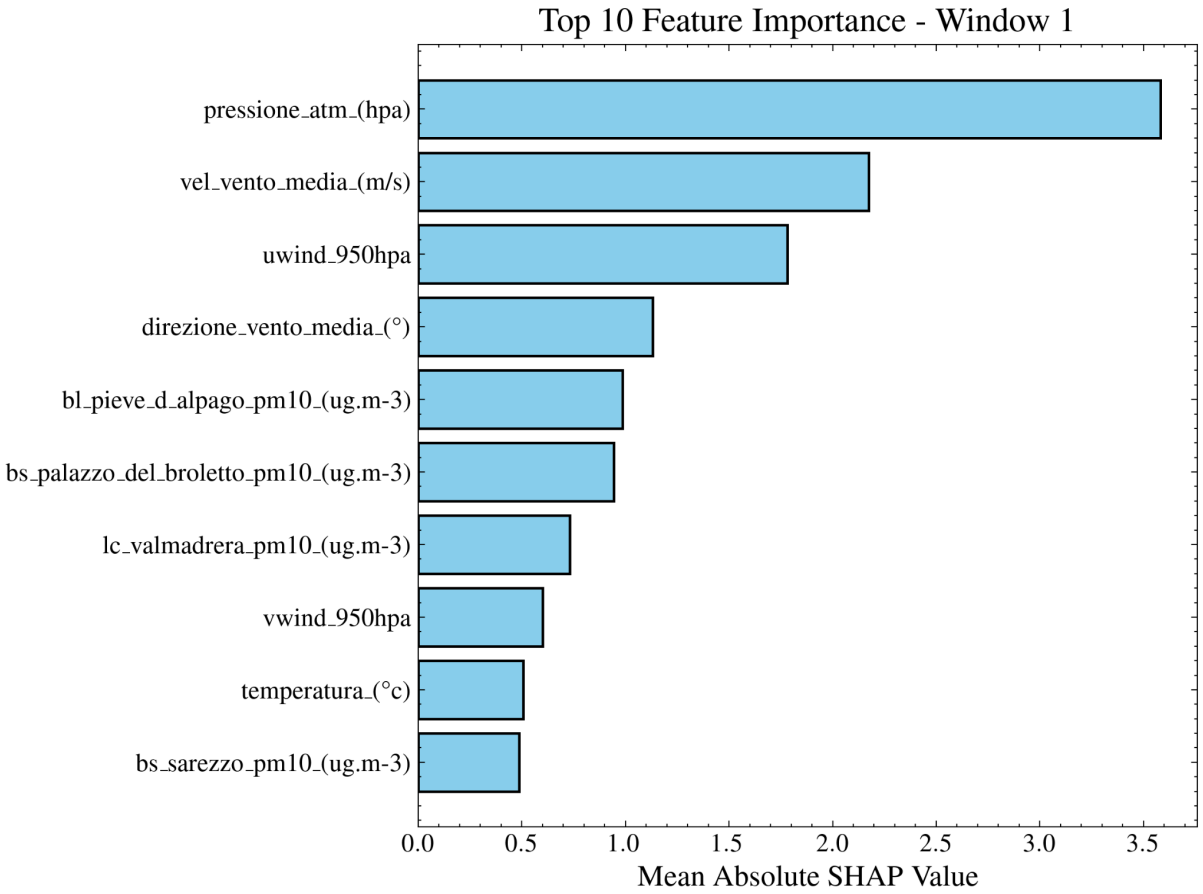


Figure 7: SHAP-based quantitative analysis showing the top 10 most influential features driving PM_{10} concentrations during Window 1 (January 2024).

Since fully understanding the quantitative SHAP analysis is impractical for a human analyst, we employ LLMs to translate these insights into natural language and infer the dynamics of PM_{10} concentrations, generating a comprehensive textual report for each of the twelve windows. In *Figure 8*, we present the outputs produced by the open-source LLM `Mistral 7B Instruct v0.2`, alongside those generated by the proprietary `GPT-4.1`, analyzing Window 1 (January 2024) and its pollutant drivers. While both LLMs produce broadly similar results, `GPT-4.1`'s output tends to be more detailed and informative, providing a deeper understanding of the factors influencing PM_{10} concentrations.

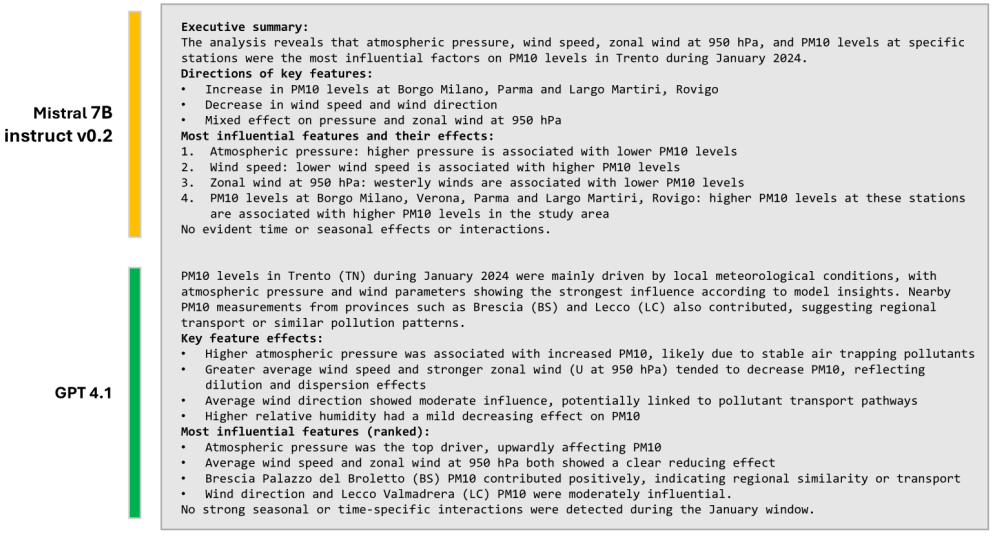


Figure 8: Comparative outputs from LLMs `Mistral 7B Instruct v0.2` and `GPT-4.1` providing explanations and inferences on PM_{10} behavior dynamics during Window 1 (January 2024).

In *Figure 9*, we illustrate a comparison between `Mistral 7B Instruct v0.2` and `GPT-4.1` in producing the final report, demonstrating how each model integrates insights from all twelve local windows into a coherent global summary.

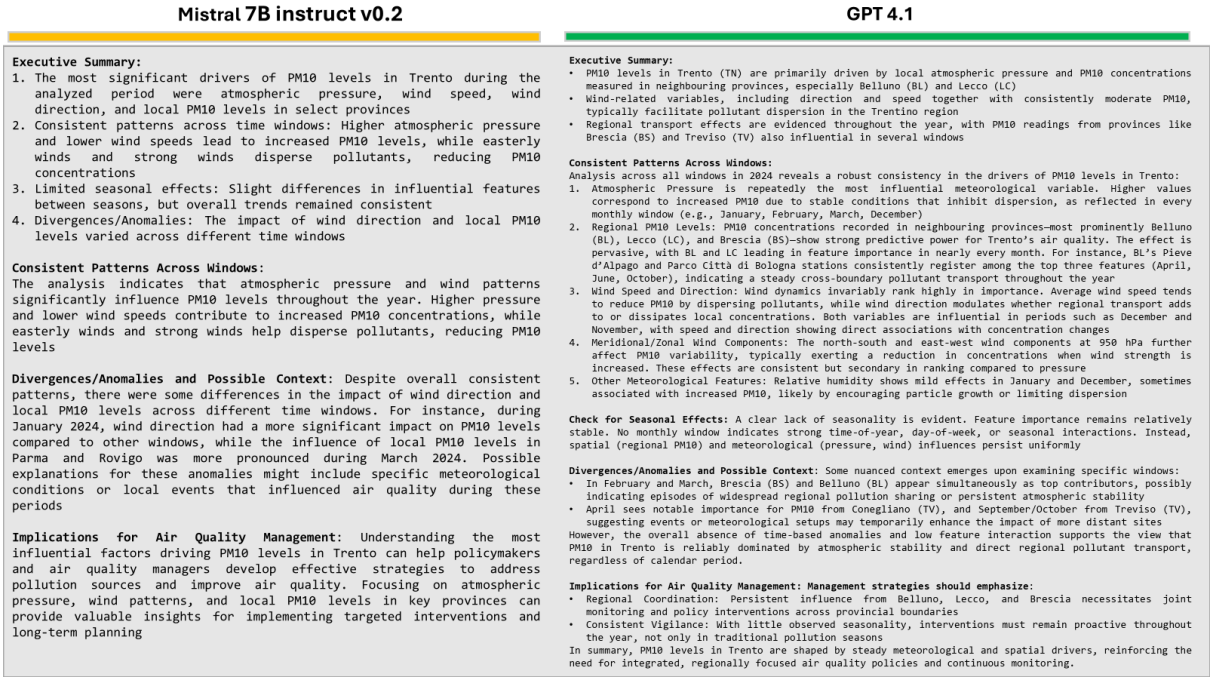


Figure 9: Comparative outputs from LLMs `Mistral 7B Instruct v0.2` and `GPT-4.1` generating a comprehensive report that explains and infers PM_{10} behavior dynamics across all twelve local windows of the 2024 test set, aggregated into a concise global annual report.

Based on the blind qualitative evaluation, `GPT-4.1` received the highest preference in 80% of cases, as reflected by seven votes for the best-quality report. These results indicate that

`GPT-4.1` outperforms `Mistral-7B-Instruct-v0.2` in generating high-quality reports according to people's judgment.

6.2. Deep learning forecasting results on the hourly dataset

Table 2 summarizes the performance across three forecasting horizons (Δt = 1,12,24 hours). Transformer-based models consistently outperformed the RNN baselines. `Timexer` and `PatchTST` emerged as the most robust architectures, achieving the lowest MAE and RMSE on the test set. Specifically, `Timexer` demonstrated superior stability on longer horizons (achieving the best RMSE for 12h and 24h), while `PatchTST` remained highly competitive, particularly in short-term accuracy. In contrast, `Samformer` exhibited overfitting, achieving near-perfect metrics on the training set (MAE ≈ 2.3) while failing to generalize to the test set (MAE ≈ 7.4).

Algorithm	Δt **(h)**	**Train**			**Test**		
		MAE	RMSE	DTW	MAE	RMSE	DTW
Crossformer	1	4.38	6.99	2986.75	6.72	10.40	3375.46
	12	4.17	6.74	2976.01	6.77	10.42	3350.09
	24	4.42	7.05	3045.13	6.92	10.60	3394.96
GRU	1	10.08	14.92	8454.23	9.48	14.25	7112.75
	12	9.86	14.79	7896.14	9.29	14.12	6518.15
	24	9.78	14.70	7912.90	9.21	14.04	6541.14
Itransformer	1	7.54	11.10	4867.11	7.30	11.07	3678.48
	12	8.70	12.69	5353.45	8.24	12.53	4041.42
	24	9.60	13.86	5453.97	8.97	13.37	4071.90
LSTM	1	11.24	15.65	9803.31	10.99	15.37	9094.05
	12	11.19	15.60	9785.67	10.94	15.31	9021.47
	24	11.19	15.60	9762.01	10.95	15.32	9043.62
Patchtst	1	3.36	5.64	2504.56	5.49	**8.37**	**2964.98**
	12	3.23	5.62	2426.98	5.94	9.16	**3181.14**
	24	3.60	6.06	2583.07	6.20	9.57	**3121.59**
Samformer	1	**2.32**	**4.33**	**1972.79**	7.35	11.10	3530.92
	12	**2.21**	**4.17**	**1951.71**	7.35	11.06	3539.40
	24	**2.61**	**4.61**	**2066.90**	7.41	11.10	3620.73
Timexer	1	5.01	7.86	3523.17	**5.46**	8.84	3135.31
	12	4.71	7.53	3299.16	**5.66**	**9.12**	3228.68
	24	5.08	8.03	3403.58	**6.00**	**9.51**	3275.78

Table 2: Results on the hourly-aggregated dataset of the PM_{10} forecast models. Best values for each metric in bold.

Despite the quantitative superiority of `Transformers`, a visual inspection of the forecasts reveals a persistent challenge: even the best-performing models struggle to accurately reconstruct sudden, high-intensity PM_{10} peaks. On one hand this behaviour is largely due to both the scale of the dataset and the chosen dimension of the tested models which did not exceed 4.4M parameters to prove the effectiveness of the approach and keep training time under a reasonable time of 8 hours. In this sense, increasing model size and training computation is very beneficial. On the other hand, this limitation is likely attributable to the boundaries of the input data: predicting acute pollution episodes driven by long-range transport (e.g., from the Po Valley) requires a broad geographical observation window and high sensor density. With poor information on incoming air masses from the south, the models lack the causal signals necessary to anticipate the onset of these rapid accumulation events. Nevertheless, all models display good overall forecasting performance both qualitatively and quantitatively, successfully capturing the general shape and temporal dynamics of the PM_{10} curve and yielding good pollutants concentration accuracy.

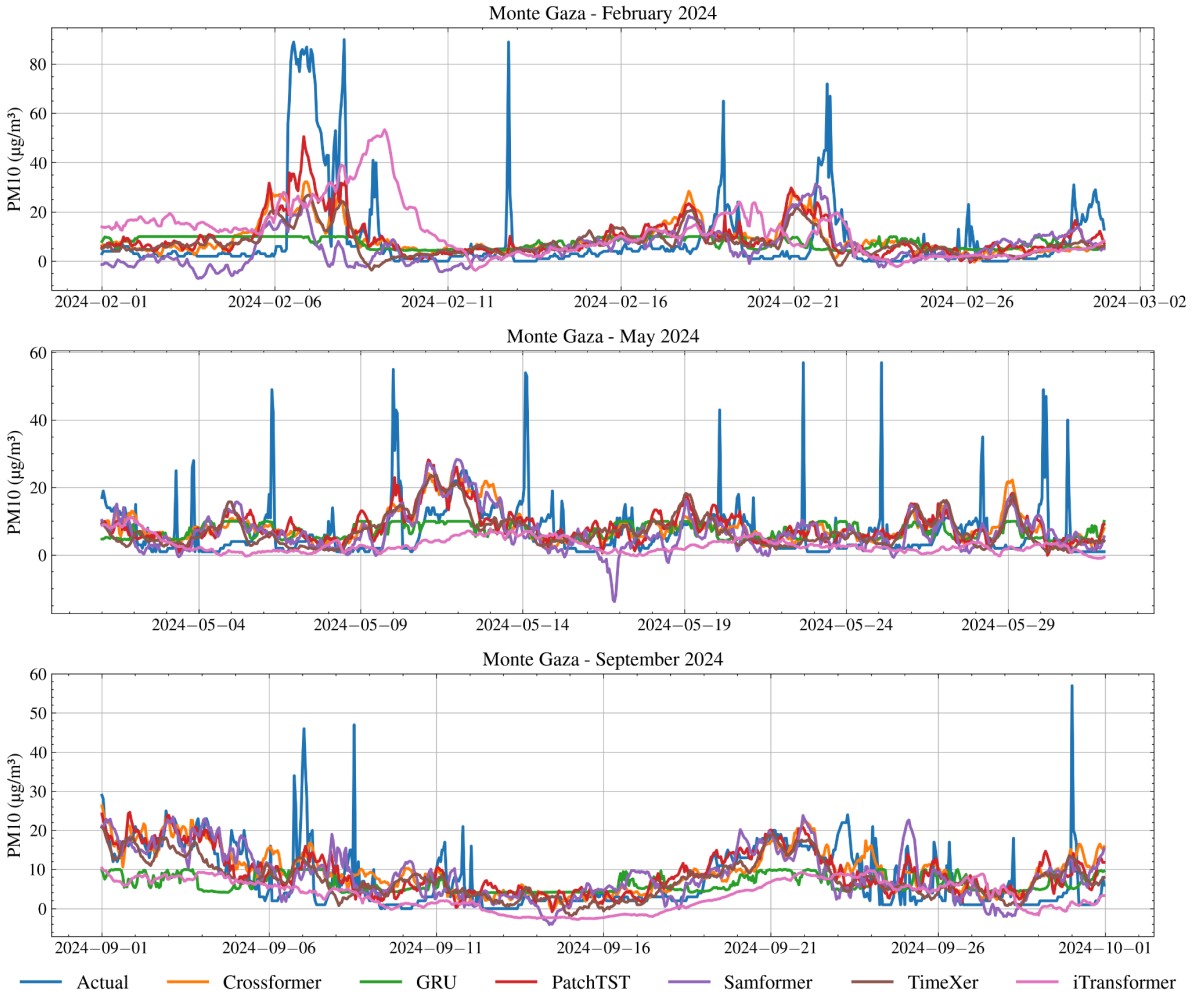


Figure 10: 2024 actual vs. predicted PM_{10} concentration profiles for APPA station Monte Gaza. Each line marks predictions for a different model architecture setting Δt to 24h.

7. Impact

Every year, APPA produces a comprehensive report analyzing pollutant concentrations in Trentino, based on both its monitoring data and external factors, such as pollutant levels in neighboring regions and meteorological conditions. Currently, these evaluations are primarily conducted by domain experts who rely on years of accumulated experience to interpret the data. In this context, our AI-based methodology offers significant potential to enhance APPA's analytical capabilities by integrating data from multiple sources and providing robust, data-driven insights. Our solution could impact APPA's work in two main ways:

1. **Explainable AI for retrospective analysis:** the XAI component can serve as a valuable reference for human analysts, helping them identify the main factors driving pollutant levels over specific historical periods. By providing clear, interpretable explanations, it supports informed decision-making and deepens understanding of complex interactions affecting air quality.
2. **Advanced deep learning for predictive forecasting:** integrating state-of-the-art deep learning architectures into an XAI pipeline remains an open research challenge, yet these methods show great potential for forecasting PM_{10} levels over medium- to

long-term horizons. By anticipating when and for how long pollutant concentrations may exceed regulatory thresholds, our approach can support APPA in taking proactive measures and planning timely interventions. From a public health perspective, timely forecasts of PM10 peaks can reduce exposure and prevent acute events, such as respiratory and cardiovascular exacerbations, especially in vulnerable groups like children, older adults and people with chronic diseases. By enabling earlier warnings and targeted mitigation measures, the system contributes directly to protecting citizens' health and improving quality of life in Trentino.

Figure 11 presents a preliminary mock-up of the web-application interface, illustrating how our approach could be deployed and used by an APPA analyst.

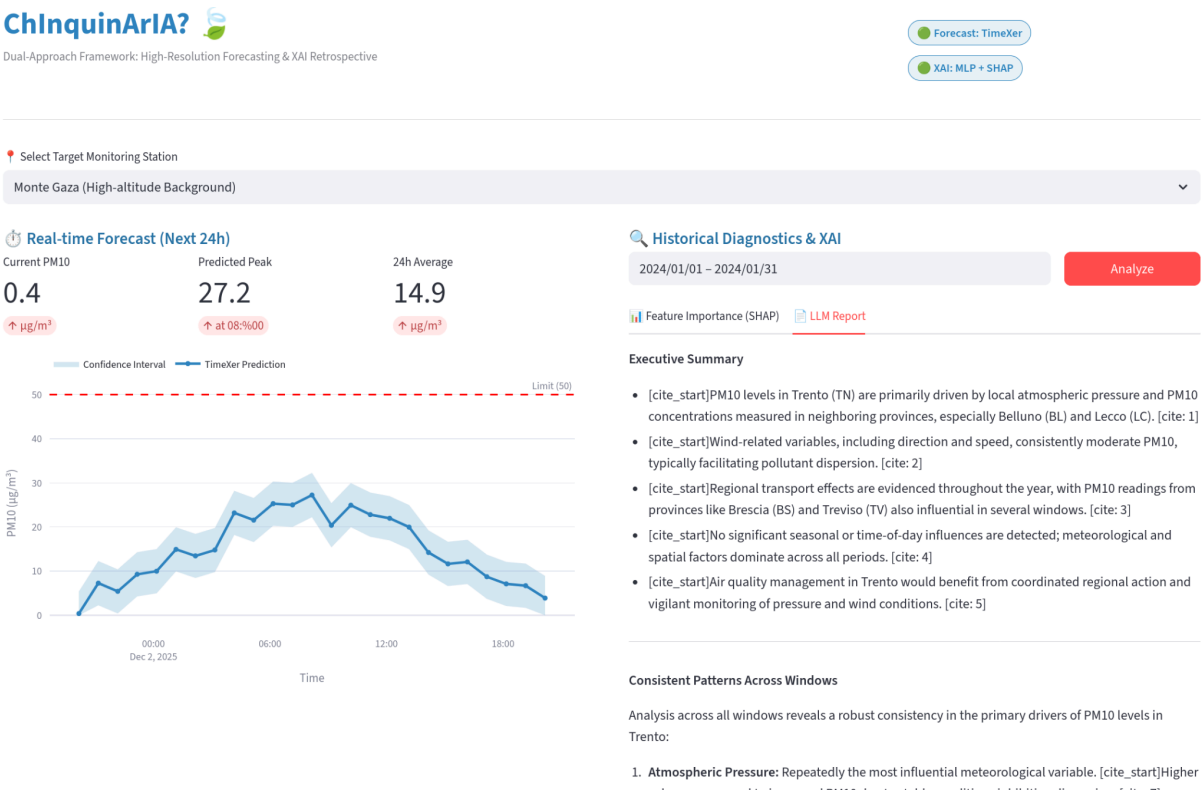


Figure 11: "ChInquinArIA?" Dashboard Mockup. A unified interface coupling high-resolution TimeXer forecasting (left) with SHAP and LLM-based explanations (right).

8. Future work and limitations

While the proposed framework demonstrates strong predictive capabilities and interpretability, several avenues remain to further enhance its robustness, spatial awareness, and physical consistency. A primary direction for improvement involves a more comprehensive characterization of the atmosphere. Currently, the model relies on a subset of ERA5 pressure levels. Future iterations should incorporate the full vertical atmospheric profile to better capture complex synoptic patterns. This is particularly critical for modeling the vertical structure of the atmosphere and improving the prediction of long-range transport events, which are often driven by upper-level flows not visible at the surface. Furthermore, we aim to integrate the extensive historical weather data already acquired from regional agencies in Veneto, Lombardia, Alto-Adige, and Trentino. Combining these dense,

ground-truth measurements with the ERA5 reanalysis will provide a more granular picture of local microclimates, provided that appropriate spatial modeling is applied to correctly propagate these local effects to the target air quality nodes.

To better address the region's complex alpine morphology, we plan to move beyond independent station modeling by adopting spatially-aware architectures, such as Graph Neural Networks (GNNs) or Spatio-Temporal Graph Convolutional Networks (ST-GCNs) (Yu et al., 2018). These models can explicitly represent the monitoring network as a graph, allowing for the inclusion of static embeddings that encode morphological and orographical knowledge, such as valley orientation and elevation (Wang et al., 2020). This would enable the model to learn how physical barriers and corridors affect pollutant dispersion. Concurrently, to increase the spatial density of observations, we intend to integrate a significantly larger number of PM_{10} monitoring stations. Since high-quality hourly data is often scarce for non-reference stations, this will require developing architectures capable of handling mixed-resolution inputs, effectively fusing abundant daily data with available hourly records to maximize information utility.

Finally, further refinements will focus on the models themselves. While our daily framework successfully leverages SHAP, the hourly Deep Learning models remain largely black boxes. Future work will investigate intrinsic explainability methods for these architectures, such as analyzing attention maps in Transformers (Chefer et al., 2021) or computing input gradients (Simonyan et al., 2013), to provide interpretable insights into high-frequency forecasts. Additionally, an exhaustive search for optimal architectural and optimization hyperparameters is expected to yield further performance gains and better convergence, surpassing the standard configurations used in this study.

9. Conclusion

This study presented a comprehensive dual-approach framework for air quality management in Trentino, successfully addressing the conflicting requirements of accurate forecasting and actionable interpretability. By constructing two robust datasets—daily and hourly—we demonstrated that Artificial Intelligence can effectively bridge the gap between complex environmental data and expert decision-making. Our experimental results highlight a clear synergy: while the XAI framework on daily data provides transparent, LLM-synthesized explanations of the physical drivers behind pollution episodes, the advanced Deep Learning architectures, particularly Transformer-based models such as `TimeXer` and `PatchTST`, deliver superior quantitative performance for medium-term forecasting. The achieved reduction in PM_{10} prediction error confirms the potential of these models to serve as reliable backbones for operational early warning systems. Notably, the value of this work extends beyond the specific case study of Trentino. The proposed methodology, combining rigorous data engineering, state-of-the-art forecasting, and automated narrative explanations, represents a generalizable paradigm that can be readily adapted to other geographical regions or environmental domains where understanding the underlying physical phenomena is just as important as predicting their future state.