Università degli studi di Milano - Bicocca

Department of Informatics, System and Communication (DISCo)

Master's Degree in Data Science

Impact of Weather Conditions on Air Quality: Integrated Analysis

Data Management Project

Davide Fabio Loreti - 865309
Carlo Pegoraro - 865329

Contents

Data sources

- https://www.weatherapi.com/my/ - Weather data source
- https://www.iqair.com/it/ - Air quality data source

API Documentation

- https://www.weatherapi.com/docs/ - WheatherAPI API Documentation
- https://api-docs.iqair.com/?version=latest - IQAIR API Documentation

Introduction

This project focuses on analyzing weather data from European capitals, obtained through the API provided by WeatherAPI.

By obtaining detailed weather information, we can gain insights into the climate conditions that influence urban environments in these large cities.

This includes parameters such as temperature, humidity and wind patterns, which are crucial to understand the broader context of air quality.

In addition to weather data, we plan to integrate air pollution information and air quality indices for these European capitals, obtained from the IQAir API. Air quality is increasingly becoming a significant concern in today's world, with urban areas, especially large cities such as European capitals, often facing challenges related to pollution from various sources, including vehicles, industrial activities and domestic heating.

Understanding the interaction between climate and air pollution is crucial for several reasons. First, weather can significantly influence the dispersion of pollutants, with factors such as wind speed and direction playing a key role in the transport of air pollutants within urban areas. Second, high levels of air pollution have well-documented negative effects on public health, contributing to respiratory diseases, cardiovascular problems and even premature mortality. By combining weather data with air quality indices, we aim to provide a comprehensive analysis that offers insights into how climate variations can directly affect air quality levels in these cities.

Furthermore, this integrated dataset will allow us to identify potential correlations between extreme weather events, such as heatwaves or heavy rainfall, and their effects on air quality. Such an understanding could inform local governments and environmental agencies, enabling them to implement targeted interventions and develop effective public health strategies.

1 Data Acquisition

As described in the introduction, the data acquisition phase was achieved using two distinct APIs:

- **WeatherAPI**: used to collect weather data for 21 European capitals. Variables obtained from this API include:
  - **City**: European capital of reference.
  - **Temperature**: current temperature in degrees Celsius.
  - **Condition**: description of the current weather condition.
  - **Humidity**: relative humidity, expressed as a percentage.
  - **Wind Speed**: wind speed in km/h.
- **IQAir AirVisual API**: used to acquire air quality data. The extracted variables include:

  - **city**: monitored European capital.
  - **state**: corresponding region.
  - **country**: corresponding country.
  - **AQI (Air Quality Index)**: air quality index according to the U.S. standard.
  - **pollutants**: detailed measurements of the present pollutants.

Since both APIs, in their free version, do not provide access to historical data, it was necessary to develop an alternative solution to ensure the temporal collection of data. To this end, an automated script was implemented which, executed daily, simultaneously acquires both weather and air quality data for each of the analyzed cities.

This strategy makes it possible to progressively build an integrated historical dataset, useful for temporal analyses and geographical comparisons. The daily storage of data allows for the examination of the evolution of atmospheric conditions and air quality over time, as well as the identification of potential correlations and recurring patterns.

2 Data Integration

Data integration represents a crucial phase of our project, as it enables the effective combination of weather information and air quality data from different sources. This process was developed through a series of methodical steps, which made it possible to create a consistent and structured dataset for subsequent analyses.

2.1 Structure of the Integration Process

The data integration process is divided into three main phases:

- **Daily data collection and harmonization**: Each day, the acquisition script simultaneously collects weather and air quality data for the 21 selected European capitals. These data are immediately subjected to quality checks to ensure consistency and reliability.

- **Creation of daily archives**: The validated data are saved in daily JSON files, where each file contains all information related to the 21 monitored cities for that specific date. The adopted hierarchical structure facilitates access to information by both city and date.

- **Temporal data integration**: Periodically, the daily files are merged into a single comprehensive JSON file, which serves as the project's historical database. This integration enables both geographical and temporal comparative analyses.

2.2 Data Quality

During the acquisition and integration phase of meteorological and air quality data for major European capitals, special attention was paid to verifying the quality of the collected information. Following the API calls to the relevant services, a validation process is triggered to ensure the completeness and semantic consistency of the received data.

For each analyzed city, the code checks for the presence of specific attributes: in the meteorological domain, it verifies the presence of city name, temperature, humidity, weather condition, and wind speed; as for air quality, it checks the availability of data regarding the city, region, country, and the AQI index along with its related pollutants.

Beyond the mere presence of fields, data type checks are performed to ensure that quantitative values are numeric, and their validity is assessed by verifying that they are positive and compatible with realistic physical limits. Only records that pass the entire validation phase are considered reliable and, therefore, included in the final dataset.

2.3 Integration Methodology

The adopted integration methodology is based on a document-oriented approach, which is particularly well-suited for managing heterogeneous data such as meteorological and air quality information. For each city and date, a structured document is created that includes both types of data, thus ensuring fast and efficient retrieval.

During this phase, special attention is paid to eliminating redundant information. For instance, city identifiers (such as the name) are stored only once within the data structure, avoiding unnecessary duplications. Similarly, redundant timestamps and other non-essential metadata are removed to optimize the dataset size.

The final structure of the integrated document follows a hierarchy of date $\rightarrow$ city $\rightarrow$ details, where "details" include both meteorological and air quality information. This organization facilitates subsequent queries based on temporal or geographical criteria.

2.4 Data Enrichment

An important component of the integration process is data enrichment, which involves adding derived or calculated information based on the original data. In our case, we implemented a categorization system for the Air Quality Index (AQI) based on the international standards of the U.S. Environmental Protection Agency (EPA).

For each recorded AQI value, a qualitative description is associated to facilitate its interpretation:

- 0–50: "Good"

- 51–100: "Moderate"
- 101–150: "Unhealthy for Sensitive Groups"
- 151–200: "Unhealthy"
- 201–300: "Very Unhealthy"
- >300: "Hazardous"

This semantic enrichment allows for the transformation of a numerical value into information that is immediately understandable and usable for user-oriented analysis and visualizations.

2.5 Challenges and Solutions

The integration of data from different APIs involved several challenges, including:

- **Temporal synchronization**: Ensuring that weather and air quality data referred to the same timestamp. This challenge was addressed by implementing a system for simultaneous data acquisition from both sources.
- **API limitations**: The restrictions imposed by the free API plans in terms of request frequency required the implementation of a controlled delay system between calls to avoid exceeding the allowed limits.
- **Handling inconsistencies**: Occasionally, some API calls may fail or return incomplete data. To handle these situations, integrity and consistency checks were implemented before integrating the data into the final dataset.

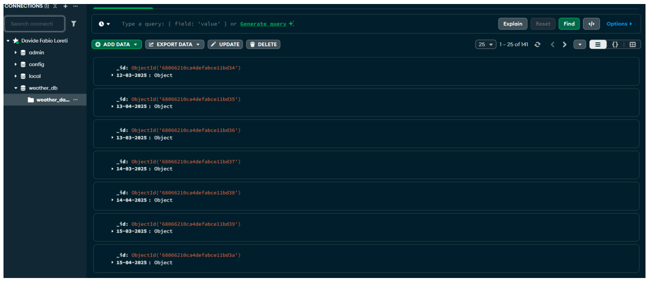
The adopted solution for these challenges was the development of a robust integration pipeline, which includes mechanisms for retrying failed calls, validating received data, and normalizing data structures to ensure homogeneity in the final dataset.

3 Data Storage

In the context of data management for our project, selecting a data storage strategy that ensures flexibility, scalability, and availability was fundamental. For this reason, we opted for the use of a NoSQL database, specifically MongoDB.

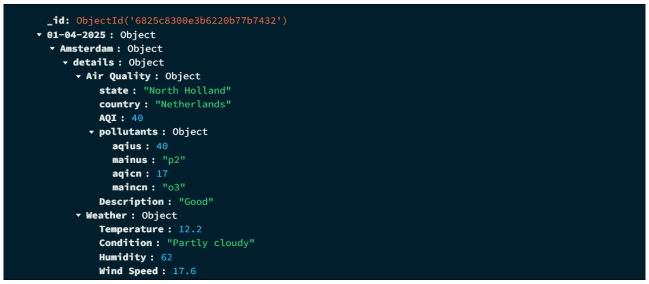
NoSQL databases differ from traditional relational databases in their ability to handle large volumes of heterogeneous and unstructured data, offering greater flexibility in data modeling and more efficient horizontal scalability. Unlike relational databases that follow the ACID properties (Atomicity, Consistency, Isolation, Durability), NoSQL databases like MongoDB are based on the BASE model:

- **Basic Availability**: the database is always available to respond to requests, even if it cannot guarantee high consistency at all times.
- **Soft State**: the state of the system may change over time, even without external input.
- **Eventual Consistency**: the system ensures that, eventually, all nodes will converge to a consistent state.



```
_id: ObjectId('68066210ca4defabce11bd34')
12-03-2025 : Object

_id: ObjectId('68066210ca4defabce11bd35')
13-04-2025 : Object

_id: ObjectId('68066210ca4defabce11bd36')
13-03-2025 : Object

_id: ObjectId('68066210ca4defabce11bd37')
14-03-2025 : Object

_id: ObjectId('68066210ca4defabce11bd38')
14-04-2025 : Object

_id: ObjectId('68066210ca4defabce11bd39')
15-03-2025 : Object

_id: ObjectId('68066210ca4defabce11bd3a')
15-04-2025 : Object
```

Figure 1: Data visualization in MongoDB



```
  _id: ObjectId('6825c8300e3b6220b77b7432')
01-04-2025 : Object
  Amsterdam : Object
    details : Object
      Air Quality : Object
        state : "North Holland"
        country : "Netherlands"
        AQI : 40
        pollutants : Object
          aqius : 40
          mainus : "p2"
          aqicn : 17
          maincn : "o3"
        Description : "Good"
      Weather : Object
        Temperature : 12.2
        Condition : "Partly cloudy"
        Humidity : 62
        Wind Speed : 17.6
```

Figure 2: Data visualization inside a document

4 Query

As part of the project, several queries were designed and executed on the MongoDB database containing daily weather and air quality data for 21 European capitals. These queries were developed to answer specific questions related to the variability of air pollution in relation to climatic conditions.

To improve usability and efficiency in interacting with the database, more complex queries were encapsulated in Python functions, making the system more *user-friendly* for the end user.

Query 1 – Air Quality in a Specific City and Date

This query returns the AQI index for a given city on a specific date:

```python
city = "Amsterdam"
date = "05-03-2025"
result = collection.find_one({date: {"$exists": True}})
if result:
    air_quality = result[date][city][’details’][’Air Quality’]
    print(f"Air Quality in {city} on {date}: {air_quality}")
```

Useful for obtaining punctual information about air pollution on a specific day.

Query 2 – Correlation between Temperature and AQI

A dedicated function computes the linear correlation between temperature and air quality for all capitals on a given day:

```python
def temp_aqi_correlation(date_str):
    query = {date_str: {"$exists": True}}
    result = collection.find_one(query)
    if result:
        l_temp = []
        l_aqi = []
        for city_name, data in result[date_str].items():
            temp = data[’details’][’Weather’][’Temperature’]
            aqi = data[’details’][’Air Quality’][’AQI’]
            l_temp.append(temp)
            l_aqi.append(aqi)
        if l_temp and l_aqi:
            correlation = np.corrcoef(l_temp, l_aqi)[0, 1]
            print(f"Correlation between Temperature and AQI: {round(correlation, 3)}")
            return correlation
```

This allows evaluating whether there are statistical relationships between climate and pollution.

Query 3 – Windiest Cities of the Day

This function retrieves the 5 cities with the highest wind speed, also displaying their AQI:

```python
def windiest_cities(date_str):
    query = {date_str: {"$exists": True}}
    result = collection.find_one(query)
    if result:
        cities_data = []
        for city_name, data in result[date_str].items():
            wind_speed = data['details']['Weather']['Wind Speed']
            cities_data.append({
                'city': city_name,
                'wind_speed': wind_speed,
                'aqi': data['details']['Air Quality']['AQI']
            })
        sorted_cities = sorted(cities_data, key=lambda x: x['wind_speed'],
     ↪ reverse=True)
        return sorted_cities[:5]
```

Useful for studying the role of wind in the dispersion of pollutants.

Query 4 – Cities with the Worst Air Quality

The following function displays the capitals with the highest AQI values on a specific date:

```python
def worst_aqi_cities(date_str, limit=5):
    query = {date_str: {"$exists": True}}
    result = collection.find_one(query)
    if result:
        cities_data = []
        for city_name, data in result[date_str].items():
            aqi = data['details']['Air Quality']['AQI']
            cities_data.append({
                'city': city_name,
                'aqi': aqi,
                'temperature': data['details']['Weather']['Temperature']
            })
        sorted_cities = sorted(cities_data, key=lambda x: x['aqi'], reverse=True)
        return sorted_cities[:limit]
```

Essential for identifying critical areas and potential environmental emergencies.

The adoption of Python functions made it possible to automate and customize the analysis, simplifying the interaction with MongoDB even for users unfamiliar with native queries. The implemented queries support exploratory analysis and provide useful insights into the relationship between climate and air quality in major European capitals.

5 Conclusions and Future Developments

The project demonstrated the feasibility and effectiveness of a systematic approach to the acquisition, integration, and analysis of data from heterogeneous sources. Through the combined use of the *WeatherAPI* and *IQAir AirVisual* APIs, it was possible to build an integrated historical dataset containing weather and air quality information for 21 European capitals.

The implementation of a data acquisition pipeline allowed us to overcome the limitations imposed by the free API versions, progressively building a historical archive of daily data. The implemented data quality process ensured the reliability of the collected information, through systematic checks on completeness, consistency, and validity.

The adoption of MongoDB as a NoSQL storage system proved particularly effective for managing semi-structured and heterogeneous data, offering the flexibility needed to support complex queries and exploratory analyses. The adopted hierarchical structure (date → city → details) facilitated access to information along both temporal and geographical dimensions.

The developed queries demonstrated the system's ability to provide meaningful insights into the relationship between weather conditions and air quality. In particular, the correlation analysis between temperature and AQI revealed interesting patterns that vary depending on the date and specific conditions, while the identification of the windiest cities and those with the worst air quality provided valuable information to understand pollutant dispersion dynamics.

Challenges Addressed and Implemented Solutions

During the development of the project, several technical challenges emerged and were addressed with innovative solutions:

- **Temporal synchronization:** the need to ensure that weather and air quality data referred to the same time was resolved by implementing a simultaneous acquisition system with temporal consistency checks.

- **API limitations:** the restrictions imposed by free API versions in terms of request frequency were handled by introducing controlled delays and retry mechanisms for failed calls.

- **Handling inconsistencies:** the implementation of data quality controls enabled the identification and filtering of incomplete or inconsistent data, ensuring the integrity of the final dataset.

- **Semantic enrichment:** the addition of qualitative descriptions for AQI values significantly improved data interpretability, transforming numerical values into user-friendly categories.

Future Developments

The project opens up several avenues for future developments that could significantly enhance its usefulness and impact:

- **Geographical and temporal expansion:** the modular architecture developed allows for easy extension of the system to include other European or non-European cities. Integrating more extensive historical data—once available through premium API versions—would enable long-term trend analyses and the identification of more robust seasonal patterns.
- **Integration of additional data sources:** the system could be enriched by integrating data from distributed IoT sensors, government monitoring stations, or satellite datasets. Including information on traffic, industrial activity, and demographics could provide a more comprehensive picture of the factors influencing air quality.
- **Predictive analysis and machine learning:** developing predictive models based on the collected historical data is a natural evolution of the project. Implementing machine learning algorithms to forecast air quality based on expected weather conditions could provide early warning tools for health and environmental authorities.
- **Advanced correlation analysis:** deepening statistical analyses through multivariate analysis techniques, space-time *clustering*, and anomaly detection could reveal hidden patterns and complex relationships between weather variables and air quality.

Impact and Practical Applications

The results of this project have potential applications in several domains:

- Public health
- Urban planning
- Scientific research

The project has shown how the integration of modern data management technologies, external APIs, and NoSQL databases can create effective information systems for analyzing complex phenomena such as the interaction between climate and air quality. The developed methodology is scalable and replicable, providing a solid framework for similar projects in other application domains.

The success of the project confirms the importance of a systematic and methodical approach to managing heterogeneous data from multiple sources, paving the way for various future developments.