International Journal of Geo-Information

MDPI

Article

Semi-Automated Reporting from Environmental Monitoring Data Using a Large Language Model-Based Chatbot

Angelica Lo Duca [1,*], Rosa Lo Duca [2], Arianna Marinelli [2], Donatella Occhiuto [2] and Alessandra Scariot [3]

[1] IIT-CNR, 56124 Pisa, Italy
[2] ARPA Lazio, 00187 Rome, Italy; rosa.loduca@arpalazio.it (R.L.D.); arianna.marinelli@arpalazio.it (A.M.); donatella.occhiuto@arpalazio.it (D.O.)
[3] University of Pisa, 56126 Pisa, Italy; a.scariot@studenti.unipi.it
* Correspondence: angelica.loduca@iit.cnr.it

Highlights

What are the main findings?

- MeteoChat, an LLM-based system optimized through fine-tuning and RAG, enables the automatic generation of environmental reports from meteorological datasets.
- The system reduces report preparation time and limits LLM hallucinations while preserving analytical accuracy and interpretability.

What are the implications of the main findings?

- By reducing human workload, the system enables timely decision-making in environmental monitoring and emergency response contexts.
- The proposed framework enhances accessibility and reproducibility in environmental data communication and reporting.

Abstract

Producing high-quality analytical reports for the environmental domain is typically time-consuming and requires significant human expertise. This paper describes MeteoChat, a semi-automatic framework for efficiently generating specialized environmental reports from heterogeneous environmental data. MeteoChat utilizes a Large Language Model (LLM) fine-tuned and integrated with Retrieval-Augmented Generation (RAG). The system's core is its plug-and-play philosophy, which separates analytical reasoning from the data source and the report's intended audience. The fine-tuning phase uses data-agnostic, parameterized question–context–answer triples defined by an environmental expert to teach the LLM domain-specific analytical logic and audience-appropriate communication styles. Subsequently, the RAG phase integrates the model with actual datasets, which are processed via an Extract–Transform–Load (ETL) workflow to generate statistical summaries. This architectural separation ensures that the same reporting engine can operate on different sources, such as meteorological time series, satellite imagery, or geographical data, without additional training. Users interact with the system via a web-based conversational interface, where responses are tailored for either technical experts (using explicit calculations and tables) or the general public (using simplified, narrative language). MeteoChat has been tested with real data extracted from the micrometeorological network of ARPA Lazio.

Keywords: data analysis; large language models; artificial intelligence; report generation; environmental monitoring

Academic Editors: Wolfgang Kainz, Huayi Wu and Zhipeng Gui

Received: 2 December 2025
Revised: 17 January 2026
Accepted: 12 February 2026
Published: 14 February 2026

1. Introduction

Producing high-quality environmental reports from heterogeneous remote sensing and meteorological datasets is a complex and time-consuming task [1]. These reports play a fundamental role in environmental monitoring by supporting timely decision-making, policy development, and public communication. However, the increasing volume, variety, and temporal depth of environmental data create significant challenges for analysts who must synthesize information accurately while communicating it effectively to users with different levels of expertise [2].

Large Language Models (LLMs) have recently emerged as powerful tools for assisting analysts in processing complex datasets and automating knowledge-intensive workflows [3]. A recent survey identified four directions for applying LLMs to environmental science: human-in-the-loop integration, the lack of domain-specific training data, standardized evaluation methods, and collaboration between AI researchers and environmental science experts [4]. Despite their potential, the direct application of LLMs in environmental monitoring remains limited by well-known issues: hallucinations, limited numerical reasoning, and difficulties ensuring explainability [5]. These limitations indicate the need for architectures that guide and constrain LLM behaviour through domain-aware reasoning patterns and reliable data integration mechanisms.

In parallel, the remote sensing community has long recognized not only the value of combining satellite data, meteorological measurements, and geospatial analytics for environmental monitoring but also the challenge of transforming raw data into meaningful decision support reports [6]. The advent of Retrieval-Augmented Generation (RAG) has opened new avenues by linking LLMs with external knowledge bases and domain-specific retrieval mechanisms [7].

This paper introduces MeteoChat, a semi-automatic framework that integrates LLM fine-tuning with RAG to generate environmental reports from both remote sensing and in situ datasets. The system follows a plug-and-play design that separates three elements: reasoning logic, data source, and target audience. This separation allows the same reporting engine to work across different environmental domains while adapting its communication style to the user's needs. Building on our previous work [8], the present paper extends the system with four main contributions:

1. *Dual-user architecture*. MeteoChat now supports both expert and non-expert audiences through two distinct fine-tuned communication styles. Expert users receive structured explanations, explicit analytical steps, and numerical tables, while general public users receive simplified narrative descriptions of environmental patterns.
2. *Generalization across datasets*. The architecture was redesigned to be independent of the underlying data modality. Through the Extract–Transform–Load (ETL) [9] and RAG workflow, MeteoChat can operate on meteorological time series, data extracted from satellite observations, or other environmental datasets without requiring additional fine-tuning.
3. *Expert-based evaluation*. Three domain experts (who are also authors of this paper) evaluated MeteoChat to assess the consistency of its reasoning, the correctness of the retrieved data, and the appropriateness of its communication style for different audiences.
4. *Enhanced report generation with automatic visualization*. A new internal visualization module now automatically generates plots and summary graphics during report compilation. This improves interpretability in expert-oriented reports and enhances readability in documents intended for non-expert audiences.

The preliminary results of this work show that MeteoChat reduces reporting preparation time, increases accessibility for non-expert users, and maintains analytical rigour for technical users.

https://doi.org/10.3390/ijgi15020080

2. Related Work

2.1. Environmental Reporting Based on Earth Observation

Several works have examined how Earth Observation (EO) contributes to environmental reporting frameworks. Andries et al. [10] analyzed the role of EO within Monitoring, Reporting and Verification (MRV) systems for land management policies and identified two main opportunities: improved spatial consistency and reduced field inspection requirements. They also identified key limitations, including heterogeneity across EO datasets, strict regulatory accuracy requirements, and gaps between policy expectations and the actual capabilities of EO indicators. Bürgin [11] showed that digital reforms have strengthened EU environmental monitoring through faster data access, more comparable indicator-based reporting, and improved verification via Copernicus data, although better data is only one of several factors influencing the Commission's overall monitoring capacity. Berger et al. [12] reviewed EO support for recent EU land-related regulations and highlighted the need for higher-resolution, more frequently updated products and for services designed around user needs. Compared to previous tools, MeteoChat introduces a plug-and-play architecture that separates analytical reasoning, data sources, and communication style, enabling the same reporting engine to operate across heterogeneous environmental datasets and to generate audience-specific reports semi-automatically.

2.2. Large Language Models in Environmental Science

The use of LLMs in environmental science has expanded rapidly, bringing both benefits, such as improved communication, and risks, such as model bias and energy consumption [5]. Raeissi and Knapen [4] categorized LLM applications in environmental science into four major groups, namely knowledge extraction, predictive modelling, question answering, and decision support, covering climate, hydrology, pollution, and biodiversity domains. Nie and Liu [13] examined the use of LLMs in decision support settings and identified two main approaches: frameworks where LLMs support human analysts by enhancing expertise and coding tasks and fully LLM-driven frameworks that aim to automate optimization processes.

Several fine-tuned environmental LLMs have recently been proposed, each addressing a specific subdomain. Ren et al. [14] developed WaterGPT, a model for water and wastewater management that supports tasks in hydrology and water resource engineering. Thulke et al. [15] introduced ClimateGPT, a climate-focused model trained on large climate corpora to support the analysis and interpretation of climate data. In the marine domain, OceanGPT [16] was designed to handle oceanographic data and assist with ocean science queries. Finally, Zhang et al. [17] introduced EnvGPT, a unified model designed to operate consistently across climate, hydrology, ecology, and soil science applications. Compared to prior models that target specific subdomains or rely primarily on text-based corpora, MeteoChat combines fine-tuning and RAG to produce audience-adapted, data-driven analyses that can support both expert assessment and public communication.

2.3. LLM-Driven and Automatic Reporting in Other Domains

Research in other scientific and technical fields provides relevant foundations for automated reporting. In radiology [18–20], using LLMs for reporting raises different concerns, including hallucinations, readability, bias and the unexplainable nature of the decision-making process [21].

Across construction and infrastructure domains, AI-based reporting systems improve efficiency, accuracy and consistency. Automated inspection tools reduce manual workload and provide reliable condition assessments [22], while multimodal frameworks like AutoRepo integrate image analysis and text generation to produce coherent site reports [23].

Broader reviews highlight that AI enhances decision-making and documentation quality across project lifecycles [24]. Although these approaches originate from domains outside GIScience, they highlight common challenges in automated reporting, such as interpretability, traceability, and the integration of heterogeneous data sources, that are also central to geospatial and environmental information systems.

2.4. Limitations of Existing Approaches and Research Gap

Most existing approaches to automated or LLM-based report generation assume a single, unified reporting workflow in which data processing, analysis, and narrative generation are tightly coupled and tailored to a specific target audience. This design limits flexibility and reusability when the same geospatial or environmental data must be communicated to different user groups, often requiring ad hoc prompt redesign or workflow reconfiguration for each audience.

While generic LLM-based systems can, in principle, be prompted to generate audience-specific outputs, this adaptation is usually achieved through ad hoc prompt engineering, which must be repeated whenever the audience or reporting context changes. Such an approach conflates data content with presentation logic, making it difficult to systematically reuse analytical results across heterogeneous audiences or datasets.

MeteoChat addresses this gap by explicitly adopting an audience-based reporting paradigm that decouples data content from communication form. Audience-specific question sets are defined once during the fine-tuning phase and can then be reused across different datasets, while the same data can be consistently communicated to multiple audiences without prompt rewriting. This separation enables both identical audiences over heterogeneous datasets and heterogeneous audiences over identical datasets, representing a key methodological contribution beyond existing approaches.

This design is consistent with data storytelling principles, in which informational content remains constant while narrative structure and emphasis vary according to the audience [25]. To the best of our knowledge, this explicit decoupling of data, analytical reasoning, and audience-adapted communication has not been systematically addressed in prior work on LLM-based environmental or geospatial reporting, positioning MeteoChat as a step toward flexible data storytelling systems within GIScience.

3. Materials and Methods

MeteoChat is a semi-automatic framework for generating environmental reports from heterogeneous remote sensing and in situ datasets. Figure 1 shows the system architecture. The architecture is organized into four main modules: communication, analysis, conversation, and reporting.

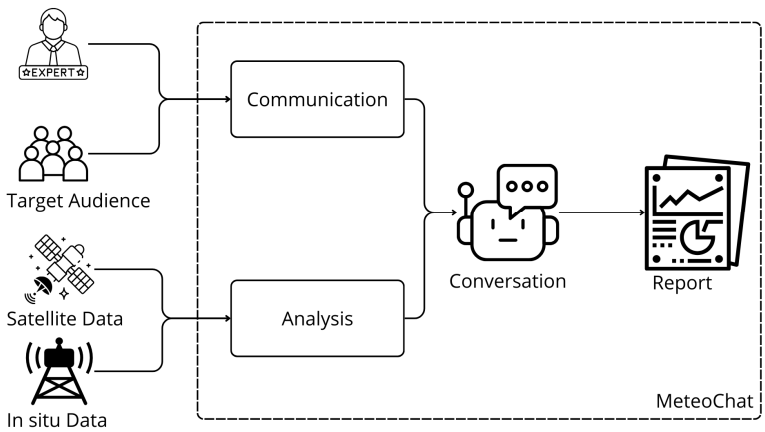


Figure 1. The MeteoChat architecture.

The combination of the analytic and communication modules follows a plug-and-play design that separates the data source and the target audience. This separation enables two flexible adaptation mechanisms. First, by modifying the set of questions and templates used during fine-tuning, the system can produce reports suited to different audiences, ranging from non-specialist users to domain experts, without altering the underlying analytical structure. Second, by replacing the dataset connected through the RAG pipeline, the same fine-tuned model can generate different reports without requiring additional training.

3.1. Communication Module

This module focuses on the audience and concerns the definition of the question and answer style and the communication format. Figure 2 shows the architecture of this module.

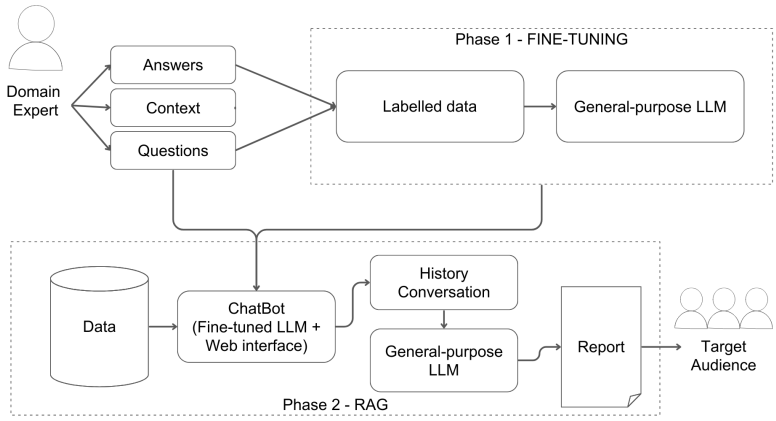


Figure 2. The communication module architecture.

A domain expert defines a curated set of question–context–answer triples, which constitute the labelled data used during the fine-tuning phase. Each triple represents a reusable analytical pattern commonly applied in environmental data analysis and is intentionally data-agnostic and parameterized so that it can be instantiated on different datasets at runtime. Specifically, each triple consists of the following. (i) *Question*: A parameterized natural language query describing a typical analytical task, where variables (e.g., year Y, station S, metric type) are placeholders rather than fixed values. (ii) *Context*: A textual description of the expected analytical procedure, explicitly defining how the result should be derived from the data (e.g., which aggregation, comparison, or selection operation to apply). (iii) *Answer*: An example response illustrating the expected structure, terminology, and level of detail of the final output, adapted to the target audience.

These triples do not contain real sensor measurements and do not encode dataset-specific values; instead, they formalize domain knowledge about how environmental analyses should be performed and how their results should be communicated. During deployment, the placeholders in the questions are instantiated with actual values and provided to the model through the RAG mechanism. An example of a question–context–answer triple is as follows. (i) *Question*: In which month of the year Y was the highest value of the examined metric recorded for station S? (ii) *Context*: Identify the maximum value in the corresponding measurement column and retrieve the month(s) associated with that value. (iii) *Answer*: The highest value was recorded in month X of year Y.

Questions and answers can be tailored to different audiences, based on two different modalities. The first modality defines specific pairs of questions and answers tailored to the target audience. For example, expert users may ask for the maximum temperature recorded by a particular sensor in a given region, expecting a detailed technical explanation that includes numerical values, methodological steps, and references to data quality. In

contrast, non-specialist users may formulate a similar question more simply, such as asking for the hottest month in that region, expecting a concise, narrative description rather than a quantitative analysis.

The second modality adapts the tone, structure, and vocabulary of the generated answer according to the user's background. For example, when an expert user requests monthly precipitation variability, the system produces a structured output with tables, explicit calculations, and domain-specific terminology. Conversely, when a non-specialist user requests the same information, the system provides an accessible narrative that highlights general patterns (e.g., "rainier periods" or "more stable months") without exposing raw statistics or technical jargon.

A generic LLM is fine-tuned on these triples to learn to interpret environmental questions, describe the analytical process, and present results with a tone and level of detail appropriate for the selected audience. The resulting model can answer specific questions using a particular tone and language but is not yet bound to any specific data source.

3.2. Analysis Module

In the second phase, the fine-tuned LLM is integrated into a Retrieval-Augmented Generation (RAG) pipeline to generate reports grounded on specific environmental datasets. This phase can be executed repeatedly on different datasets without requiring additional model retraining. Before ingestion into the RAG system, raw data undergo a deterministic Extract–Transform–Load (ETL) process. Data cleaning is performed independently for each environmental metric (e.g., temperature, precipitation, pressure) and includes: (i) the removal or flagging of missing and invalid values, (ii) timestamp normalization and alignment to a uniform temporal resolution, and (iii) the separation of records by year and monitoring station. Subsequently, validated datasets are aggregated to compute statistical indicators commonly used in environmental reporting, such as the mean, median, mode, minimum, and maximum. These computations are performed using conventional analytical tools and manually verified, ensuring numerical correctness before any interaction with the LLM.

The cleaned and aggregated data are then converted into structured textual documents that summarize validated statistical indicators for a given metric, station, and time period. This transformation is fully deterministic and does not introduce additional interpretation or linguistic variability: each document directly reflects precomputed numerical results following a fixed template. As a consequence, document quality depends on upstream data validation rather than on natural language generation, and no separate linguistic evaluation is required for this step.

Each textual document was segmented into chunks before indexing. Chunk size is a key design parameter in RAG systems, as it influences the trade-off between contextual completeness and retrieval noise. Following a preliminary quantitative study [8], chunk sizes ranging from 500 to 10,000 characters were tested, and a size of approximately 1000 characters was selected as a compromise between response quality and computational efficiency. Chunks were embedded using a dense embedding model and stored in a vector database. The selected embedding model was chosen for its robustness on structured technical text and its compatibility with the adopted RAG infrastructure. At query time, relevant chunks were retrieved based on semantic similarity and provided as external context to the fine-tuned LLM.

3.3. Conversation Module

The conversation module manages user interaction with MeteoChat via a web-based conversational interface (Figure 3). This module represents the operational layer of the

framework, where the fine-tuned model and the RAG engine converge to support semi-automatic reporting. The goal is to enable experts and non-experts to query environmental datasets in natural language and obtain accurate, contextually relevant answers.

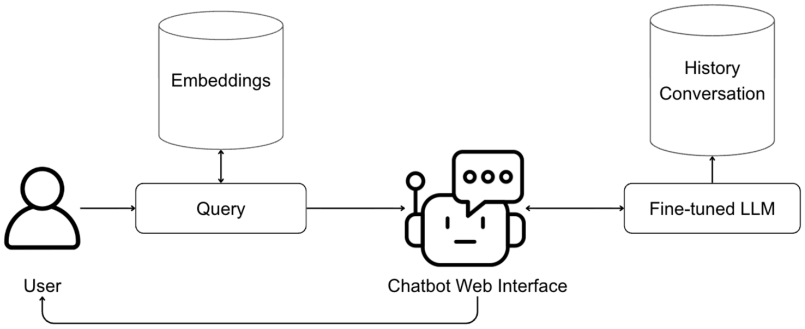


Figure 3. The conversation module architecture.

The fine-tuned model is based on a GPT-4o LLM deployed through Azure OpenAI services. While different LLMs could be integrated into the proposed architecture, their comparative evaluation is beyond the scope of this study. Fine-tuning was performed using the set of question–context–answer triples described before, without modifying the model architecture. Training was conducted for a limited number of epochs with default optimization parameters provided by the platform, as the goal was stylistic and structural alignment rather than learning numerical computation. The RAG pipeline consists of a vector store populated with embedded textual summaries. Each document is chunked and embedded offline, while retrieval at query time is performed using cosine similarity. Retrieved chunks are injected as external context to the fine-tuned LLM, ensuring that all numerical values originate from deterministic preprocessing.

The chatbot is implemented as a browser-based interface that provides a real-time conversational environment. Two interface versions are available: expert and non-specialist modes. Both interfaces share the same visual layout (Figure 4), where the central portion of the page displays the evolving conversation, and the lower portion contains a text input field for submitting new queries and a dedicated button for downloading the final report.

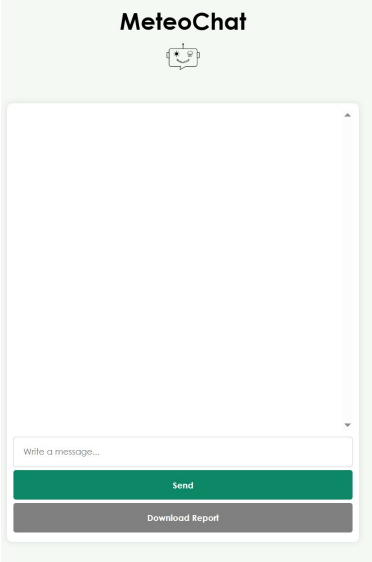


Figure 4. The interface of the mobile version of the chatbot, where the conversation area is located in the centre, allowing users to view received and sent messages. At the bottom, there is a text field where users can type their questions, as well as a "Send" button for sending messages and a "Download Report" button for downloading reports.

In addition to the list of available questions for different types of audiences, the distinction between the two lies in the set of instructions (prompt template) provided to the fine-tuned model. In expert mode, responses maintain a precise technical tone, include explicit reasoning steps, incorporate statistical terminology, and refer directly to the values retrieved from the RAG module. The model is directed to present data in tables, perform calculations transparently, and organize its answers into clearly defined sections such as Data, Calculations, and Conclusion. The prompt template uses a one-shot prompting technique [26], which describes how to transform raw values into a table and explicitly provides step-by-step reasoning instructions. The prompt template for experts contains four main parts: the context, which defines the relevant information extracted from the RAG mechanism; the question to answer; the detailed steps for performing the analysis; and the layout formatting rules, such as the style for formatting tables, if any. The following snippet of code shows the main elements of the prompt template:

```
CONTEXT:
Consider the following context: {context}
QUESTION:
Answer the following question: {question}

ANALYSIS:
When performing calculations:
    - Show calculations clearly as plain text, step by step.
    - Example:
      Average Pressure 2019 = (1011.43 + 1021.14 + 1018.56 + 1013.03 + ...)
      / 12 = 12,175.59 / 12 = 1014.63 mbar
    - Use section labels such as:
      Data:
      Calculations:
      Conclusion:

LAYOUT:
Format any list of data as a **table** with clear headers.
Example transformation:
    - January: 3.5 mm
    - February: 7.2 mm

Should be formatted as:

| Month | Max Precipitation (mm) |
|----------|------------------------|
| January | 3.5 |
    | February | 7.2 |

The final answer must be clear, structured, and easy to read in plain
text format.
```

The non-specialist mode adopts simplified language, narrative elements, shorter sentences, and a focus on interpretability rather than numeric detail. The prompt template for non-expert users uses the role-prompting technique [11], which assigns a specific role to the model (e.g., "You are an environmental expert…") to guide its tone, style, and response mode. In this configuration, the chatbot behaves like a meteorologist, explaining environmental data to a general audience.

The prompt template for non-specialist users contains four main parts: the role, which defines the specific role the model must assume; the context, which describes the relevant information extracted from the RAG mechanism; the question to answer; and the analysis, which explains the analysis performed. The following snippet of code shows the main elements of the prompt template:

```
ROLE
You are a meteorologist who explains environmental data to a general
audience. Your goal is to transform technical information into short,
engaging, and clear narratives that highlight meaningful trends or changes.

Principles to follow:
    - Tone: conversational, informative, and vivid---but not exaggerated.
    - Focus on clarity and insight more than storytelling flair.
    - Use simple metaphors or imagery only if they help understanding
    (avoid overly poetic language).
    - Keep answers concise and fact-driven.
    - Connect the data to real-world implications or everyday experience
    when possible.
    - Avoid technical jargon and excessive numbers---summarize trends in
    plain language.

                        CONTEXT:
                        Consider the following context: {context}

                        QUESTION:
                        Answer the following question: {question}

                        ANALYSIS:
                        When performing calculations, explain them
briefly and simply.
```

The main difference between the two prompt templates is that the non-specialist prompt template uses a simplified, narrative style to make environmental data easier to understand, avoiding technical jargon and numerical detail, while the user expert prompt template employs a structured, analytical approach to ensure accuracy and methodological precision.

When the user submits a query, the system retrieves the relevant textual segments from the RAG index, forwards them to the fine-tuned model together with the question, and returns an answer tailored to the previously selected audience. The fine-tuned model ensures that each answer preserves the expected analytical structure learned during training. In contrast, the RAG module ensures that all numerical results are derived from the underlying dataset. During the conversation, all exchanges are stored in chronological order. This logging step is essential because the entire interaction becomes the basis for the final report.

3.4. Reporting Module

Once the user activates the download button, the system retrieves the conversation history and invokes a general-purpose LLM to generate a coherent document structure. Report generation was fully automated using GPT-4o, deployed through Azure OpenAI services. The model first generates a title and an abstract that align with the conversation's content. In expert reports, the title and abstract use technical terminology, whereas in non-specialist reports, they prioritize clarity and accessibility.

Each user question is converted into a section header, and the corresponding chatbot answer becomes a paragraph placed below it. For expert-oriented reports, the system additionally inserts summary tables that extract key quantitative values from the conversation and, when available, embeds plots generated through the internal visualization module. For non-specialists, the report's structure is designed to maintain simplicity: summary tables may be omitted when deemed too technical, and emphasis is placed on short, descriptive explanations rather than exhaustive statistics. At the end of the process, the system generates a complete document containing the title, abstract, keywords, question–answer sections, and any supplementary elements such as plots or conclusions.

In terms of prompts, the main difference between expert and non-specialist users lies in the specification of the intended audience. For example, the prompt used to generate a conclusion for a non-specialist audience is as follows:

```
Write a conclusion for a general audience about this conversation
(3--4 sentences):{conversation}.
```

The conversation contains the overall conversation between the user and the chatbot. For expert users, instead, the prompt for conclusions is as follows:

```
Generate the conclusion for the report, addressed to expert users, based
on this conversation (3--4 sentences): {conversation}.
```

Although the difference between the two formulations is subtle, it nonetheless yields consistently effective, context-appropriate outputs.

4. The ARPA Lazio Micrometeorological Monitoring Network

To evaluate MeteoChat, we conducted a case study focusing on the operational meteorological network of ARPA Lazio, which provides high-frequency in situ measurements from multiple monitoring stations.

4.1. The Network

In 2012, ARPA Lazio established a micrometeorological monitoring network designed to support advanced air quality assessment and pollutant dispersion forecasting across the Lazio region. The network consists of nine fixed monitoring stations (Table 1).

Table 1. The stations at ARPA.

Station Code	Location
AL001	Tor Vergata (Roma)
AL002	Latina
AL003	Tenuta del Cavaliere (Roma)
AL004	Castel di Guido (Roma)
AL005	Rieti
AL006	Frosinone Military Airport
AL007	Boncompagni (Roma)
AL008	Viterbo Military Airport
AL009	Ceprano

4.2. Measured Parameters

The parameters made available annually are listed in Table 2, along with notes and corresponding units of measurement.

Table 2. Legend of meteorological parameters and corresponding units of measurement.

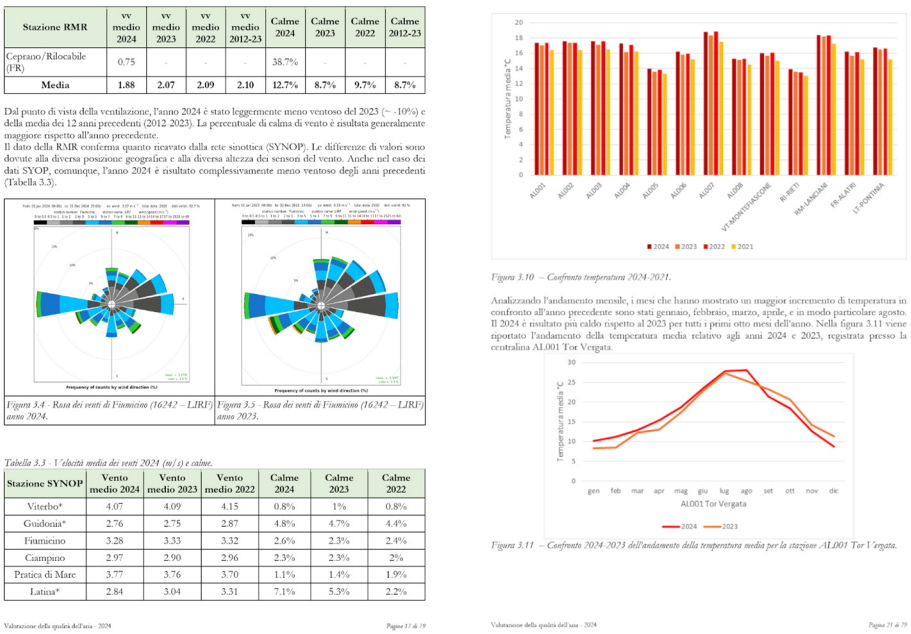
Attribute	Description
Station code	station code in the form of AL00X
Date/Time	yyyymmdd_hhii (year, month, day, hour, minutes)
Temperature	°C (the value −9,999,900 indicates the absence of data)
Relative humidity	% (the value −9,999,900 indicates the absence of data)
Wind speed	m/s (the value −9,999,900 indicates the absence of data)
Wind direction	direction from the north (the value −9,999,900 indicates the absence of data)
Precipitation	cumulative mm (the value −9,999,900 indicates the absence of data)
Atmospheric pressure	mbar reduced to sea level (the value −9,999,900 indicates the absence of data)
Global radiation	W/sqm (the value −9,999,900 indicates the absence of data)
Albedo	W/sqm (the value −9,999,900 indicates the absence of data)
Atmospheric infrared	W/sqm (the value −9,999,900 indicates the absence of data)
Terrestrial infrared	W/sqm (the value −9,999,900 indicates the absence of data)
Net radiation	W/sqm (the value −9,999,900 indicates the absence of data)

In addition to the series of traditional meteorological parameters made public, the ARPA Lazio micrometeorological network plays a crucial role in collecting an extended set of derived micrometeorological parameters. These include, for example, absolute air humidity, mixing ratio, and dew point temperature. The total dataset collected by the network comprises 55 distinct parameters. These data, which are fundamental for advanced studies in atmospheric research and environmental modelling, are not published routinely but are made available, upon specific request, to researchers and interested parties for scientific purposes.

The monitoring network is designed to characterize boundary-layer meteorology and to investigate how micro-scale atmospheric processes influence the transport and dispersion of air pollutants across heterogeneous environments within the region. More details about the ARPA Lazio micrometeorological network are available in Appendix A.

4.3. Reporting

Every year, ARPA Lazio publishes the document "Air Quality Assessment" on its institutional website (https://www.arpalazio.it/web/guest/ambiente/aria/pubblicazioni, accessed on 10 February 2026). This document presents the analysis of last year's data to verify compliance with regulatory limits in municipalities in the Lazio region. The document includes a chapter dedicated to the region's meteorological analysis, as this factor is closely linked to air quality (Figure 5). MeteoChat fits into this context of report generation and statistical data analysis, serving as a potential resource for speeding up and/or validating statistics generated with other tools or even for expanding analyses for public reporting.



Stazione RMR	vv medio 2024	vv medio 2023	vv medio 2022	vv medio 2012-23	Calme 2024	Calme 2023	Calme 2022	Calme 2012-23
Ceprano/Rilocabile (FR)	0.75	-	-	-	38.7%	-	-	-
Media	**1.88**	**2.07**	**2.09**	**2.10**	**12.7%**	**8.7%**	**9.7%**	**8.7%**

Dal punto di vista della ventilazione, l'anno 2024 è stato leggermente meno ventoso del 2023 (~ -10%) e della media dei 12 anni precedenti (2012-2023). La percentuale di calma di vento è risultata generalmente maggiore rispetto all'anno precedente.
Il dato della RMR conferma quanto ricavato dalla rete sinottica (SYNOP). Le differenze di valori sono dovute alla diversa posizione geografica e alla diversa altezza dei sensori del vento. Anche nel caso dei dati SYOP, comunque, l'anno 2024 è risultato complessivamente meno ventoso degli anni precedenti (Tabella 3.3).

Figura 3.4 - Rosa dei venti di Fiumicino (16242 – LIRF) anno 2024.

Figura 3.5 - Rosa dei venti di Fiumicino (16242 – LIRF) anno 2023.

Tabella 3.3 - Velocità media dei venti 2024 (m/s) e calme.

Stazione SYNOP	Vento medio 2024	Vento medio 2023	Vento medio 2022	Calme 2024	Calme 2023	Calme 2022
Viterbo*	4.07	4.09	4.15	0.8%	1%	0.8%
Guidonia*	2.76	2.75	2.87	4.8%	4.7%	4.4%
Fiumicino	3.28	3.33	3.32	2.6%	2.3%	2.4%
Ciampino	2.97	2.90	2.96	2.3%	2.3%	2%
Pratica di Mare	3.77	3.76	3.70	1.1%	1.4%	1.9%
Latina*	2.84	3.04	3.31	7.1%	5.3%	2.2%

Valutazione della qualità dell'aria - 2024 Pagina 31 di 79

Figura 3.10 – Confronto temperatura 2024-2021.

Analizzando l'andamento mensile, i mesi che hanno mostrato un maggior incremento di temperatura in confronto all'anno precedente sono stati gennaio, febbraio, marzo, aprile, e in modo particolare agosto. Il 2024 è risultato più caldo rispetto al 2023 per tutti i primi otto mesi dell'anno. Nella figura 3.11 viene riportato l'andamento della temperatura media relativo agli anni 2024 e 2023, registrata presso la centralina AL001 Tor Vergata.

Figura 3.11 – Confronto 2024-2023 dell'andamento della temperatura media per la stazione AL001 Tor Vergata.

Valutazione della qualità dell'aria - 2024 Pagina 33 di 79

Figure 5. An example of a chapter dedicated to meteorological analysis.

5. Implementation

This section describes how MeteoChat was fine-tuned and integrated with the meteorological data from ARPA Lazio.

5.1. Fine-Tuning

The fine-tuning phase did not rely directly on ARPA data but used ARPA as the reference domain for defining the analytical reasoning patterns. As a model to fine-tune, we selected OpenAI GPT-4o. The focus of this work is not on benchmarking different LLMs but on evaluating the proposed fine-tuning and RAG-based reporting framework. For this reason, alternative LLMs were not systematically evaluated, as the model choice is orthogonal to the architectural contribution of MeteoChat. Three of the authors acted as domain experts and created a structured set of question–context–answer triples derived from the kinds of queries commonly applied to the ARPA dataset, such as identifying the month with the highest value of a variable, calculating annual averages, determining the largest intra-monthly variation, or examining differences between consecutive years. During this phase, two separate question sets were prepared: one for domain experts, who require detailed explanations and methodological clarity (Table A2), and one for general users, who benefit from simplified, narrative-style responses (Table A3).

The questions related to the domain expert set include two complementary types of queries. The first type focuses on single-station analyses, where statistical indicators are computed and interpreted for an individual monitoring site. The second type introduces inter-station comparison queries, which require reasoning across multiple geographically distributed stations to identify spatial extremes, quantify differences in cumulative indicators, or assess spatial heterogeneity within the monitoring network.

The differences between the two types of audiences were incorporated into the answer templates, enabling the model to modulate its tone, level of detail, and structure according to the final audience. The fine-tuning produced two models, one for each audience type.

5.2. RAG

Once each fine-tuned model was available, it was integrated with the actual ARPA Lazio data through the RAG layer. The RAG phase used the real measurements from the

ARPA network and served as the operational data layer. The raw ARPA datasets were first processed through a dedicated ETL pipeline. This phase involved cleaning the original CSV files, removing anomalous or missing values, standardizing timestamps, and creating separate annual files for each metric, such as temperature, precipitation, and atmospheric pressure. After cleaning, descriptive statistics including the mean, minimum, maximum, median, and mode were computed to create compact and information-rich summaries for each year.

Before building the RAG index, an analysis was conducted to determine the optimal chunk size for text segmentation [8]. Chunk sizes from 500 to 10,000 characters were tested. The analysis showed that extremely small chunks often fragmented contextual information, while excessively large chunks slowed retrieval and introduced noise. Intermediate chunks provided the best compromise, with the system achieving stable scores around level 3 across most configurations. As a result, a chunk size of approximately 1000 characters was selected for the ARPA case study, ensuring both contextual adequacy and retrieval efficiency. The cleaned and statistically enriched ARPA documents were then indexed as embeddings with the optimized chunk size and made accessible to the chatbot.

5.3. Simulating Interaction

The conversation in the chatbot interface was evaluated through a series of controlled interactions designed to simulate the typical workflow of an environmental expert querying the system. In this phase, the goal was to observe how the model behaved during real exchanges, the clarity and coherence of its answers, and the extent to which it could support semi-automatic reporting. This phase was evaluated through a series of interactions replicating typical user behaviour across two distinct audiences: domain experts and general public users. Although fine-tuning was performed using questions combining multiple stations (see Appendix B for details), as a representative example, the focus of simulation was on station AL007 because the behaviour is similar for the other stations. Future work will also focus on questions involving multiple stations.

Domain expert. We defined a representative example consisting of three sequential queries. Figure 6 shows a snapshot of the MeteoChat interface with a user question and the chatbot's answer. The first question concerned temperature extremes: "Which month of the year 2020 had the highest temperature value recorded for station AL007?" The chatbot identified the month associated with the maximum recorded temperature and returned a concise yet technically grounded explanation. The response was produced in approximately five seconds.

The second question, "What is the average annual value of precipitation in 2014 for station AL007?", required the system to process a larger number of measurements. The chatbot returned the computed annual average together with a brief description of the underlying calculation. This answer took around 28 s to produce, reflecting the increased complexity of the aggregation involved.

The third question focused on intra-annual variability in atmospheric pressure: "In which month of the year 2023 does the pressure show the greatest discrepancy between its maximum and minimum values for station AL007?" The chatbot correctly identified the month with the largest pressure range and summarized the associated values. The system produced this answer in approximately 6 s. Across all expert-oriented tests, the chatbot maintained a coherent technical tone, referred explicitly to the underlying values, and provided explanations consistent with professional analytical practice.

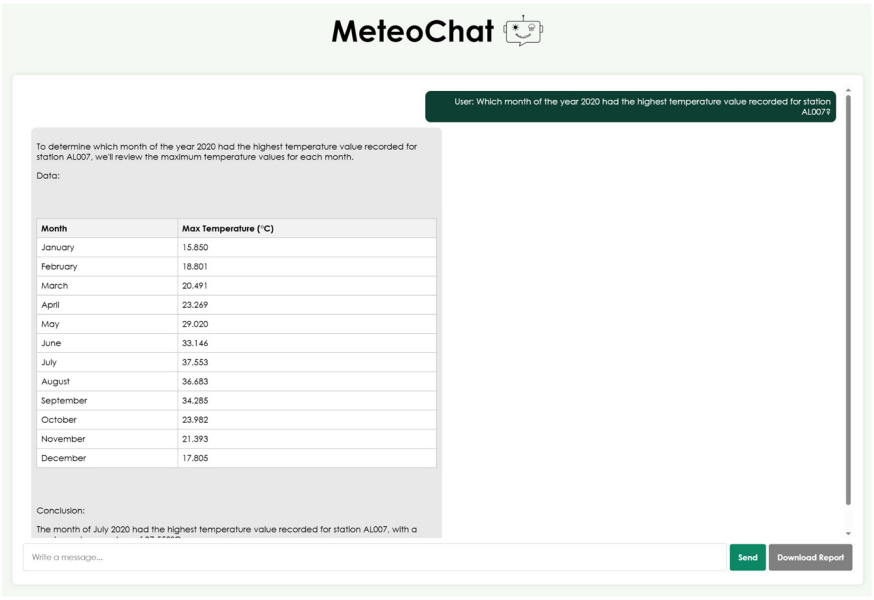


Figure 6. An example of a conversation in which the domain expert asks the chatbot one question, highlighted in dark green, concerning which month of 2020 recorded the highest temperature for station AL007.

Non-specialist user. A parallel conversational evaluation was conducted using the non-specialist modality, which employs simplified language, avoids technical terminology, and focuses on narrative clarity. In a representative interaction (Figure 7), the user asked the following: "How did the year 2020 go in terms of temperatures at station AL007?" Instead of providing numerical extremes or step-by-step calculations, the chatbot generated an accessible summary of the year's overall temperature pattern. The answer highlighted the warmest and coolest periods and provided a qualitative interpretation of seasonal behaviour, avoiding specific numerical values unless strictly necessary for clarity. The chatbot emphasized concepts such as "warmer months", "cooler periods", and an "overall stable trend", offering a clear and intuitive overview for non-specialist readers.

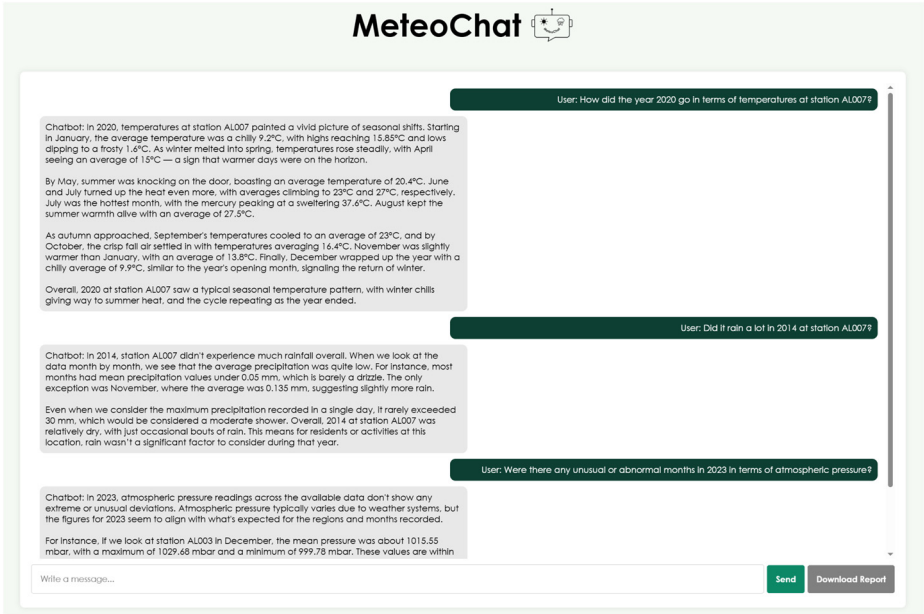


Figure 7. An example of a conversation in which the non-specialist user asks the chatbot three questions, highlighted in dark green, while the chatbot's answers, in light grey, are narrative and use simple terms.

The second question tested the chatbot's ability to describe precipitation conditions. When the user asked, "Did it rain a lot in 2014 at station AL007?", the model provided a narrative assessment of whether 2014 was relatively wet or dry. It summarized rainfall distribution using accessible expressions such as "a rather rainy year" or "rainfall within normal ranges", rather than presenting aggregated monthly values or more technical indicators.

The third question demonstrated how the system communicates variability without relying on explicit statistical measures. The user asked the following: "Were there any unusual or abnormal months in 2023 in terms of atmospheric pressure?" The chatbot identified the month showing the most noticeable fluctuations and described it qualitatively as "more unstable", "showing more evident variations", or "less uniform than the other months", avoiding references to maximum–minimum discrepancies or intra-month variability metrics.

5.4. Report Downloading

At the end of the experimental session, both users downloaded a report by simply clicking the corresponding button in the MeteoChat interface. Once activated, the system extracted the whole conversation, organized it into a structured layout, and compiled a Microsoft Word document. The generation process required a short waiting period, after which the file became available in the browser's download folder.

The resulting report reproduced the interaction exactly as it occurred during the experiment. The first page contained an automatically generated title, abstract, and keywords derived from the conversation's content. From this point onward, the report's structure and tone diverged depending on whether the chatbot was in expert or non-specialist mode.

In the expert version, each question asked during the experiment was converted into a section heading, followed by a detailed paragraph containing the chatbot's answer (Figure 8). These answers retained their technical character, including explicit references to maximum and minimum values, annual averages, and month-to-month variations. The report also incorporated visual elements: when numerical values were present in the conversation, corresponding plots were generated and embedded directly into the document to support technical interpretation. The final page contained a summary table listing, for each question, a concise keyword descriptor and the key numerical value extracted from the answer. This structure provided a compact analytical overview suitable for professional use.

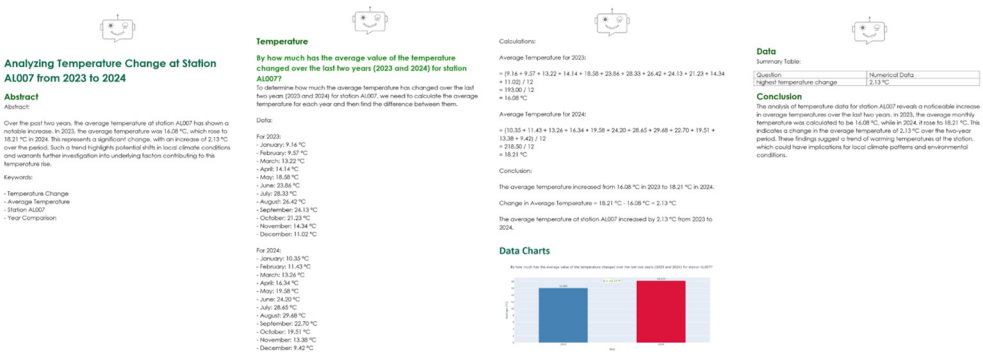


Figure 8. An example of the final report (intended for expert users) in which the left-hand page shows the title, abstract, and keywords generated by the general-purpose LLM. In contrast, the middle pages show the user's question and the chatbot's answer, followed by a graph generated from the data contained in the response. The last page, on the right, includes a summary table and conclusion, generated by a general-purpose LLM.

In the non-specialistic version, the same sequence of questions was transformed into a more accessible narrative (Figure 9). The generated abstract was shorter and written in plain language, and each section presented the chatbot's responses in descriptive terms rather than through explicit statistics. Instead of numerical values, the text emphasized qualitative assessments such as identifying warmer or cooler periods, wetter or drier years, or months showing more evident variability. Visual elements were included only when they improved readability without introducing technical complexity. No summary table was added because listing numerical indicators was considered inappropriate for this audience.

Figure 9. An example of the final report (intended for standard users). The left-hand page shows the title, abstract, and keywords generated by the general-purpose LLM. The middle pages present two user's questions—one about temperature changes during 2024 and another about the months with the highest values of precipitation—followed by the chatbot's answers and the graphs generated from the data contained in the responses. The right-hand page contains the conclusion, also generated by a general-purpose LLM.

5.5. Evaluation

The evaluation of MeteoChat aims to assess the quality, robustness, and practical usefulness of the generated environmental reports in an expert-oriented setting. In addition, it focuses on its role as a support tool for human-driven reporting, not as a replacement for expert-authored documents. The framework is designed to operate alongside human analysts to accelerate the reporting process while preserving expert oversight; therefore, the evaluation does not include a direct comparison with fully human-authored reports.

Three separate conversations were designed, each corresponding to a distinct set of questions. Each conversation was repeated three times, yielding a total of nine independent conversational sessions. For each session, MeteoChat generated a separate report, resulting in nine reports to be evaluated. One of the authors (who is also a domain expert) manually verified the correctness of the generated outputs by independently recomputing the corresponding indicators using Microsoft Excel, confirming that the numerical results were correct. Consequently, the subsequent expert evaluations do not assess output correctness but rather reflect each evaluator's perception of report quality, usefulness, and adequacy.

5.5.1. Evaluation Design and Limitations

Three independent domain experts (three of the authors), each with experience in environmental monitoring and data interpretation, were asked to evaluate the nine reports individually. While this represents a methodological limitation due to potential bias, this choice was dictated by the highly specialized nature of the domain and the need for evaluators with: (1) deep knowledge of ARPA Lazio's operational workflows, (2) technical expertise in meteorological data interpretation, and (3) familiarity with institutional reporting requirements. To mitigate potential bias, evaluators were asked to assess reports independently without discussion and provide detailed written justifications for their

ratings. We acknowledge that future work should include external independent evaluators from other environmental agencies to strengthen validity.

The evaluation was conducted on 45 annotated items (nine reports × five criteria) rated independently by the three domain experts using a 10-point Likert scale. Table 3 shows the evaluation criteria and Table 4 the average rating for each evaluator across the different criteria.

Table 3. Evaluation parameters and their definitions.

Parameter	Description
Perceived Accuracy	Degree to which numerical values in the report are coherent with the true values measured by the micrometeorological station; indicates reliability and technical soundness.
Informational Completeness	Extent to which the report covers key indicators, trends, and relevant analytical elements; also reflects the system's ability to convey results through text, graphics, or other media.
Expositive Clarity	Readability and fluency of the text, including grammatical correctness, structural coherence, and ease of comprehension.
Terminological Coherence/Technical Style	Appropriateness and consistency of technical terminology, reflecting adherence to disciplinary standards and the production of unambiguous, professionally usable descriptions.
Operational/Institutional Utility	Practical usefulness of the report for supporting day-to-day environmental monitoring activities and for informing institutional decision-making processes.

Table 4. The aggregated results for the evaluation.

Criteria	Evaluator 1	Evaluator 2	Evaluator 3
Perceived Accuracy	10	10	9
Informational Completeness	8	10	9
Expositive Clarity	10	8	9
Terminological Coherence/Technical Style	10	10	9
Operational/Institutional Utility	8	8	8.5

5.5.2. Inter-Rater Agreement Analysis

Inter-rater reliability metrics (Krippendorff's Alpha (ordinal), Intraclass Correlation Coefficient ICC(2,k), weighted Cohen's Kappa, Kendall's W, and percent exact agreement) were used to quantify the variability in expert judgments on report quality. The resulting values were low (Krippendorff's Alpha = 0.07; ICC(2,k) = 0.18; weighted Kappa = 0.04–0.16; Kendall's W = 0.38, non-significant), indicating substantial variability among evaluators. This outcome reflects the interpretative nature of the rated dimensions, which include perceived usefulness, data aggregation adequacy, and operational relevance. Accordingly, agreement-based metrics were used to describe the dispersion of expert evaluations.

In this context, the low agreement metrics do not indicate evaluator inconsistency but rather reflect three distinct expert perspectives shaped by different analytical priorities (data granularity vs. narrative clarity), varying operational contexts (field monitoring vs. institutional reporting), and disciplinary backgrounds (meteorology vs. data science vs. policy). A qualitative analysis of evaluator comments (see Section 5.5.3) reveals that disagreements were systematic rather than random, with Evaluator 3 consistently rating lower due to concerns about monthly aggregation, while Evaluators 1–2 prioritized institutional usability.

5.5.3. Qualitative Analysis of Evaluator Comments

To contextualize the quantitative results, the evaluators also provided free-text comments. Evaluator 1 emphasized the reliability of the numerical outputs and their consistency with the results obtained using standard tools such as Excel. The evaluator explicitly noted that values related to temperature, pressure, and precipitation fully matched expected benchmarks, confirming robustness and correctness at the data level.

Evaluator 2 focused on the system's potential for institutional and operational use. The comments underline that the reports offer precise and accurate data, integrate textual explanations and graphical elements, and are broadly suitable for institutional contexts. At the same time, the evaluator suggested that minor refinements could further improve usability, reflecting a forward-looking assessment oriented toward deployment rather than fundamental system limitations.

Evaluator 3 expressed a more critical perspective, particularly in relation to tasks performed on monthly aggregated data. The evaluator noted that, for certain analyses, such as counting missing temperature measurements, aggregation leads to a loss of detail and results that are not directly comparable with those obtained from raw time series analysis in tools like Excel. This feedback does not question the internal consistency of MeteoChat but instead highlights a mismatch between the analytical granularity required by specific tasks and the level of data aggregation adopted in the experiment.

5.5.4. Comparison with a Baseline

A direct comparison with a non-fine-tuned baseline model was conducted in previous work using the same reporting tasks and evaluation criteria [8]. That comparison showed that MeteoChat, through fine-tuning and structured question–answer logic, produces more detailed, transparent, and report-like outputs than a general-purpose LLM, particularly in its explicit explanation of analytical reasoning steps. While this paper emphasizes expert judgement rather than direct baseline comparison, these prior results provide complementary evidence of the added value introduced by the proposed framework.

6. Conclusions and Future Work

This paper presented MeteoChat, a semi-automatic framework that leverages a fine-tuned LLM combined with RAG to support the generation of environmental reports. MeteoChat was tested on meteorological data from the ARPA Lazio network, and the nine expert-oriented reports produced were consistently judged positively by the three evaluators, with scores concentrated in the upper range of the Likert scale. Although statistical reliability metrics did not show significant agreement, the qualitative assessments indicate that the reports were coherent, readable, and adequate for operational use.

Several limitations emerged. The evaluation relied on only three domain experts, which restricts the robustness of agreement analyses. The current implementation also operates exclusively on hourly CSV data, limiting validation across other data modalities. Moreover, the ETL pipeline still requires a preliminary manual cleaning phase, which constrains full automation and may reduce scalability when integrating larger or noisier datasets.

In addition, the current implementation relies on a vector-based RAG mechanism, which is effective for contextualizing and explaining precomputed indicators but may be less suitable for strictly quantitative queries, such as counting rainy days, calculating standard deviations, or identifying specific timestamps for maximum values. In MeteoChat, this limitation is partially mitigated by the ETL pipeline, which computes all numerical statistics upstream and validates them before providing them to the LLM. Future work will investigate hybrid architectures that integrate structured querying approaches, such

as Text-to-SQL, to improve accuracy, traceability, and reproducibility in expert-oriented reporting scenarios.

In addition, current evaluation focused on pipeline correctness and output reliability, while audience-specific language adaptation and comparative user studies are left for future work.

Finally, the need for a manual data-cleaning phase within the ETL pipeline currently constitutes a bottleneck to the system's full automation and scalability. To address this, future work will investigate integrating AI Agent architectures that can autonomously handle data ingestion, detect anomalies, and perform corrective preprocessing without human intervention. By utilizing an agentic workflow to manage the ETL process, the system could more effectively transition from processing static CSV files to handling large-scale, real-time, or noisier environmental datasets.

Future developments will focus on expanding the system to include satellite-derived remote sensing products, enabling the assessment of multi-source environmental scenarios. Enhancing the data ingestion pipeline with AI-driven mechanisms for autonomous anomaly detection and correction will further increase robustness and reduce preprocessing effort. Additional testing with more evaluators, diversified user questions, and broader environmental contexts will be essential to strengthen the system's reliability and demonstrate its applicability to a wider range of monitoring and reporting tasks. Overall, these directions aim to consolidate MeteoChat as a flexible and scalable tool for automated environmental reporting.

Author Contributions: Conceptualization, Angelica Lo Duca and Rosa Lo Duca; Related Work, Angelica Lo Duca; Materials and Methods, Angelica Lo Duca and Alessandra Scariot, with contributions from the other authors; Software, Alessandra Scariot; The ARPA Lazio Micrometeorological Network, Rosa Lo Duca, Arianna Marinelli and Donatella Occhiuto; Experiments: all; Conclusion, all; Writing—Original Draft Preparation, Angelica Lo Duca, with contributions from all the authors; Writing—review and editing, all. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available in the ARPA Lazio repository at https://www.arpalazio.it/rete-micro-meteorologica (accessed on 10 February 2026).

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

Appendix A

Each site in the ARPA micrometeorological monitoring network is configured with a standardized instrumentation setup that includes an ultrasonic anemometer (USA-1 Scientific), a tipping-bucket rain gauge (VRG-101), a thermo-hygrometer (HMP-45AC), a soil temperature profiler (QMT-103), a net radiometer (CNR1), and a soil heat flux plate (HFP01), together with standard sensors for temperature, relative humidity, atmospheric pressure, and precipitation. All stations comply with the World Meteorological Organization (WMO) siting and measurement quality requirements. The spatial distribution of the nine monitoring stations within the Lazio region is illustrated in Figure A1.

Figure A1. Location of micrometeorological network stations (image from Google Earth).

A detailed overview of the instrumentation installed at the AL007 micrometeorological station is reported in Table A1. The station is equipped with a set of high-precision sensors that capture key atmospheric variables relevant to environmental and air quality monitoring. While some instruments are standard across all stations in the network, AL007 includes additional components that enable advanced measurements, including three-dimensional wind characterization, energy balance estimation, and high-accuracy thermo-hygrometric profiling. The table summarizes the main instruments, their measurement principles, operational ranges, and output characteristics, providing a concise reference for understanding the data types and quality used in this paper.

Table A1. Instrumentation installed at AL007 micrometeorological station.

Instrument	Model	Type/Purpose	Main Features	Operating Range/Output
Ultrasonic Anemometer	USA-1 Scientific	3-axis ultrasonic anemometer for wind speed and direction	• No moving parts, low maintenance • High precision, wind tunnel-calibrated • Sampling frequency: up to 50 Hz (2D), 30 Hz (3D) • Wind speed range: 0–60 m/s	• Temperature: −40 °C to +70 °C • Optional sensor heating
Rain Gauge	VRG-101	Weighing tipping-bucket rain gauge for liquid/solid/mixed precipitation	• Measures cumulative precipitation (mm) and intensity (mm/h) • Rim heating to prevent ice/snow buildup • Intelligent control to minimize evaporation	—

Table A1. *Cont.*

Instrument	Model	Type/Purpose	Main Features	Operating Range/Output
Thermo-Hygrometer	HC2A-S3	Digital/analogue probe for temperature and relative humidity	• Accuracy: ±0.8%RH, ±0.1 °C (10–30 °C) • Humidity range: 0–100%RH (non-condensing) • Wind resistance up to 50 m/s (mesh filter) • Digital UART + dual analogue outputs	• Temperature: −50 °C to +100 °C • Analogue output standard: 0–1 V = −40. . .60 °C, 0–1 V = 0–100%RH
Four-Component Radiometer	CNR1	Measures shortwave and longwave radiation; computes energy balance	• Two pyranometers (CM3) for incident/reflected shortwave • Two pyrgeometers (CG3) for longwave • Integrated Pt1000 sensor • Heater for dew/frost prevention • Analogue mV outputs proportional to irradiance	• Response time: 18 s (95%) • Temperature: −40 °C to +70 °C
Barometer	PTB110	Atmospheric pressure measurement	• Capacitive silicon sensor • High stability: ±0.1 hPa drift/year	• Analogue voltage: 0–2.5 V or 0–5 V • Frequency output: 500–1100 Hz

Figure A2 shows the typical micrometeorological station on the ARPA Lazio network. The micrometeorological network continuously records micrometeorological parameters at a 30 s sampling frequency and transmits them to a central server for archiving and processing. The raw data undergo multiple validation phases. The traditional meteorological parameters (temperature, relative humidity, wind speed and direction, precipitation, pressure, global radiation, albedo, atmospheric infrared, terrestrial infrared, and net radiation) are then published and made publicly available on the ARPA Lazio website at Rete micro-meteorologica—ARPA Lazio.

Figure A2. A typical micrometeorological station on the ARPA Lazio network.

Appendix B

This appendix contains the list of questions used to fine-tune the model for domain experts (Table A2) and for non-specialist users (Table A3).

Table A2. The list of questions used to fine-tune the LLM for domain experts.

Question	Context	Answer
In which month of the year Y was the highest value of the examined metric recorded for station S?	Find the maximum value in the measurement column and return the corresponding month(s).	The highest value was recorded on month X of year Y.
By how much do the maximum and minimum values of the examined metric differ in year Y for station S?	Find the maximum and minimum values and calculate the difference.	The maximum and minimum values differ by X units.
How many times did the metric drop below value X at station S in year Y?	Count how many measurements are below 0 °C.	The temperature dropped below 0 °C X times in year Y.
What is the average annual value of the examined metric in year Y for station S?	Calculate the average of all measured values.	The average annual value of the parameter in year Y is X.
What was the most frequent value (mode) of the examined metric in year Y for station S?	Find the most frequently occurring value.	The most frequent value of the parameter in year Y was X.
In year Y, how many measurements of the examined metric were not recorded due to technical issues with station S?	Calculate missing measurements assuming 48 per day for 366 days in a leap year.	The number of measurements not recorded due to technical issues was X in year Y.
In which month of the year Y was the lowest value of the examined metric recorded for station S?	Find the minimum value and return the corresponding month(s) that correspond to it.	The lowest value was recorded on month X of year Y.
By how much has the average value of the examined metric changed over the last two years (Y1 and Y2) for station S?	Calculate the average for each year, then compute the difference.	The average value changed by X units between 2023 and 2024.
When ordering the dataset of the examined metric in ascending order, what is the median value for year Y for station S?	Sort the values and calculate the median.	The median value of the parameter in year Y is X.
In which month of year Y does the examined metric show the greatest discrepancy between its maximum and minimum values for station S?	For each month, calculate the difference between the maximum and minimum values. Return the month with the highest difference.	The month with the greatest discrepancy is X.
What is the average annual wind speed in year Y for each station?	You are a data analyst showing data to the general public. Consider the parameter wind speed. The data consist of half-hourly measurements for station Z in year Y. Consider a table consisting of one column: the measured wind speed values in m/s. Count how many valid values are present in the table. Sum all the valid wind-speed values. Apply the following formula: total sum of values/number of valid values = result. Return this result as the output.	The average annual wind speed in year Y for station Z is X m/s.

Table A2. *Cont.*

Question	Context	Answer
What is the total annual (cumulative) precipitation in year Y for station Z?	You are a data analyst showing data to the general public. Consider the parameter precipitation. The data consist of half-hourly measurements for station Z in year Y. Consider a table consisting of one column: the measured precipitation values in millimetres (mm). Sum all the remaining values in the table. The result represents the total annual (cumulative) precipitation in mm. Return this result as the output.	The total annual precipitation in year Y for station Z is X mm.
On which day of year Y did station Z record its absolute maximum daily precipitation?	You are a data analyst showing data to the general public. Consider the parameter precipitation. The data consist of half-hourly measurements for station Z in year Y. Consider a table consisting of two columns: date (day) and measured precipitation values in millimetres (mm). For each calendar day, sum all the precipitation values of that day to obtain one daily total per day. Then, among the daily totals, find the highest value. The day corresponding to this highest daily total represents the day with the absolute maximum precipitation. Return this day as the output.	The absolute maximum daily precipitation in year Y at station Z was recorded on day X.
How many rainy days (with precipitation greater than 1 mm) were recorded in year Y at station Z?	You are a data analyst showing data to the general public. Consider the parameter precipitation. The data consist of half-hourly measurements for station Z in year Y. Consider a table consisting of two columns: date (day) and measured precipitation values in millimetres (mm). For each calendar day, sum all the precipitation values of that day to obtain one daily total per day. Count how many days have a daily total strictly greater than 1 mm. This count represents the number of rainy days. Return this count as the output.	In year Y, station Z recorded X rainy days with precipitation greater than 1 mm
What is the average annual temperature in year Y for station Z?	You are a data analyst showing data to the general public. Consider the parameter temperature. The data consist of half-hourly measurements for station Z in year Y. Consider a table consisting of one column: the measured temperature values in °C (for example, column T). Count how many valid values are present in the table. Sum all the valid temperature values. Apply the following formula: total sum of values/number of valid values = result. The result represents the annual average temperature. Return this result as the output.	The average annual temperature in year Y for station Z is X °C.

Table A2. *Cont.*

Question	Context	Answer
On which day of year Y was the maximum temperature recorded for station Z?	You are a data analyst showing data to the general public. Consider the parameter temperature. The data consist of half-hourly measurements for station Z in year Y. Consider a table consisting of two columns: date (day and time) and measured temperature values in °C. Among the remaining values, identify the highest temperature value. Then retrieve the date corresponding to this highest value. Return this date as the output.	The maximum temperature in year Y for station Z was recorded on day X.
On which day of year Y was the minimum temperature recorded for station Z?	You are a data analyst showing data to the general public. Consider the parameter temperature. The data consist of half-hourly measurements for station Z in year Y. Consider a table consisting of two columns: date (day and time) and measured temperature values in °C. Among the remaining values, identify the lowest temperature value. Then retrieve the date corresponding to this lowest value. Return this date as the output.	The minimum temperature in year Y for station Z was recorded on day X.
Which of the nine weather stations recorded the highest temperature in year Y?	You are a data analyst showing data to the general public. Identify the maximum temperature recorded in year Y for each of the nine weather stations, then compare these values and select the highest one.	The weather station Z recorded the highest temperature in year Y.
What is the maximum difference in cumulative precipitation among all weather stations in year Y?	You are a data analyst showing data to the general public. To answer the question, compute the cumulative precipitation for each weather station in year Y, then calculate the difference between the highest and the lowest cumulative values.	The maximum difference in cumulative precipitation among the stations in year Y is X units.
Which weather station recorded the highest and the lowest wind speed in year Y?	You are a data analyst showing data to the general public. To answer the question, find the maximum and minimum wind speed recorded at each station in year Y, then identify the highest value among the maxima and the lowest value among the minima.	The highest wind speed was recorded at station Z1, while the lowest wind speed was recorded at station Z2 in year Y.
Which of the nine weather stations shows the largest daily thermal excursion?	You are a data analyst showing data to the general public. Using daily data only, compute the daily thermal excursion (daily maximum minus daily minimum temperature) for each of the nine weather stations, then identify the station with the largest excursion.	Station Z shows the largest daily thermal excursion.

Table A3. The list of questions used to fine-tune the LLM for non-specialist users.

Question	Context	Answer
How did the average value of the examined metric change month by month in year Y?	Calculate the monthly averages for the selected metric and list or plot them in chronological order.	In year Y, the average value of the examined metric changed month by month as follows: [. . .]
Which month had the highest value of the examined metric in year Y?	Identify the maximum value in the dataset and return the month in which it occurred.	The highest value was recorded in month X of year Y.

Table A3. *Cont.*

Question	Context	Answer
Compare the average values of the examined metric across years Y1, Y2, and Y3.	Compute annual averages for each selected year and compare them.	The average values for the years were: Y1 = X1, Y2 = X2, Y3 = X3.
What was the difference between the highest and lowest recorded values of the examined metric in year Y?	Find the maximum and minimum values within the year and subtract them.	The difference between the highest and lowest values is X units.
How much was recorded each month for the examined metric in year Y?	Sum or average (depending on metric type) the monthly values in chronological order.	Monthly recorded values for the year Y are: X1 for M1, X2 for M2, . . ., X12 for M12.
In which month was the examined metric the highest?	Identify the month with the greatest measured value.	The metric was highest in month X.
Compare the total annual amount of the examined metric between years Y1 and Y2.	Compute yearly totals (or yearly averages) for both years and compare them.	Year Y1 recorded X1 units, while year Y2 recorded X2 units.
What is the range between the month with the highest value and the month with the lowest value in year Y?	Find the maximum and minimum monthly values and compute the difference.	The range between the highest and lowest months is X units.
Show the distribution of the examined metric for year Y.	Analyse all values recorded during the selected year and summarise them statistically.	The distribution of values for year Y is as follows: [. . .]
How did the average value of the examined metric evolve throughout year Y?	Compute monthly averages and compare them chronologically.	The average value evolved as follows: [. . .]
Which year had the highest average value for the examined metric?	Compute the annual average for all available years and select the maximum.	Year Y recorded the highest average value.
Compare the minimum and maximum values of the examined metric in year Y.	Extract the lowest and highest recorded values and present them side by side.	In year Y, the minimum was X1 and the maximum was X2.
Show the variability of the examined metric during a certain season (e.g., summer).	Analyse values within the selected months and summarize variability.	During the specified season, values varied as follows: [. . .]
Compare multiple metrics (e.g., metric A, metric B, metric C) month by month in year Y.	Retrieve monthly values for each metric and present them together for comparison.	The comparison for year Y is as follows: [. . .]
Did the examined metric show unusually high or low values at station S in year Y?	Identify outliers or values outside expected ranges for that station/year.	Yes/No. The unusual values were observed in the following months: [. . .]
How did year Y compare to previous years in terms of the examined metric at station S?	Compare annual averages or totals with historical values.	Year Y was (higher/lower/similar) compared to previous years.
Were there any abnormal months in year Y for the examined metric at station S?	Detect anomalies based on thresholds, deviations, or statistical irregularities.	Yes/No. Abnormal months include: [. . .]

References

1. Kolk, A. Evaluating corporate environmental reporting. *Bus. Strategy Environ.* **1999**, *8*, 225–237. [CrossRef]
2. Vourvachis, P.; Woodward, T. Content analysis in social and environmental reporting research: Trends and challenges. *J. Appl. Account. Res.* **2015**, *16*, 166–195. [CrossRef]
3. Gu, Y.; You, H.; Cao, J.; Yu, M.; Fan, H.; Qian, S. Large Language Models for Constructing and Optimizing Machine Learning Workflows: A Survey. *ACM Trans. Softw. Eng. Methodol.* **2025**. [CrossRef]
4. Raeissi, M.M.; Knapen, R. Applications of Generative Large Language Models in Environmental Science: A Systematic Review. *Adv. Environ. Eng. Res.* **2025**, *6*, 028. [CrossRef]

5. Rillig, M.C.; Ågerstrand, M.; Bi, M.; Gould, K.A.; Sauerland, U. Risks and Benefits of Large Language Models for the Environment. *Environ. Sci. Technol.* **2023**, *57*, 3464–3466. [CrossRef] [PubMed]
6. Pasetto, D.; Arenas-Castro, S.; Bustamante, J.; Casagrandi, R.; Chrysoulakis, N.; Cord, A.F.; Dittrich, A.; Domingo-Marimon, C.; El Serafy, G.; Karnieli, A.; et al. Integration of satellite remote sensing data in ecosystem modelling at local scales: Practices and trends. *Methods Ecol. Evol.* **2018**, *9*, 1810–1821. [CrossRef]
7. Juhasz, M.; Dutia, K.; Franks, H.; Delahunty, C.; Mills, P.F.; Pim, H. Responsible Retrieval Augmented Generation for Climate Decision Making from Documents. 2024. Available online: https://arxiv.org/abs/2410.23902 (accessed on 10 February 2026).
8. Duca, A.L.; Duca, R.L.; Scariot, A. MeteoChat: A Fine-tuned and RAG-based LLM for Semi-Automatic Report Building in Environmental Monitoring. In *Computer-Human Interaction Research and Applications. CHIRA 2025*; Communications in Computer and Information Science; Springer: Cham, Switzerland, 2025.
9. Vassiliadis, P. A Survey of Extract–Transform–Load Technology. *Int. J. Data Warehous. Min. IJDWM* **2009**, *5*, 1–27. [CrossRef]
10. Andries, A.; Murphy, R.J.; Morse, S.; Lynch, J. Earth Observation for Monitoring, Reporting, and Verification within Environmental Land Management Policy. *Sustainability* **2021**, *13*, 9105. [CrossRef]
11. Bürgin, A. Modernization of Environmental Reporting as a Tool to Improve the European Commission's Regulatory Monitoring Capacity. *JCMS J. Common Mark. Stud.* **2021**, *59*, 354–370. [CrossRef]
12. Berger, K.; Foerster, S.; Szantoi, Z.; Hostert, P.; Foerster, M.; Van De Kerchove, R.; Vancutsem, C.; Schweitzer, C.; Masolele, R.; Reiche, J.; et al. Evolving Earth observation capabilities for recent land-related EU policies. *Land Use Policy* **2025**, *158*, 107749. [CrossRef]
13. Nie, Q.; Liu, T. Large language models: Tools for new environmental decision-making. *J. Environ. Manag.* **2025**, *375*, 124373. [CrossRef] [PubMed]
14. Ren, Y.; Zhang, T.; Dong, X.; Li, W.; Wang, Z.; He, J.; Zhang, H.; Jiao, L. WaterGPT: Training a Large Language Model to Become a Hydrology Expert. *Water* **2024**, *16*, 3075. [CrossRef]
15. Thulke, D.; Gao, Y.; Pelser, P.; Brune, R.; Jalota, R.; Fok, F.; Ramos, M.; van Wyk, I.; Nasir, A.; Goldstein, H.; et al. ClimateGPT: Towards AI Synthesizing Interdisciplinary Research on Climate Change. 2024. Available online: https://arxiv.org/abs/2401.09646 (accessed on 10 February 2026).
16. Bi, Z.; Zhang, N.; Xue, Y.; Ou, Y.; Ji, D.; Zheng, G.; Chen, H. OceanGPT: A Large Language Model for Ocean Science Tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*; Ku, L.-W., Martins, A., Srikumar, V., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2024; pp. 3357–3372. [CrossRef]
17. Zhang, Y.; Lin, S.; Xiong, Y.; Li, N.; Zhong, L.; Ding, L.; Hu, Q. Fine-tuning large language models for interdisciplinary environmental challenges. *Environ. Sci. Ecotechnol.* **2025**, *27*, 100608. [CrossRef] [PubMed]
18. Leonardi, G.; Portinale, L.; Santomauro, A. Enhancing radiology report generation through pre-trained language models. *Prog. Artif. Intell.* **2024**. [CrossRef]
19. He, Z.; Wong, A.N.N.; Yoo, J.S. Radiology report generation using automatic keyword adaptation, frequency-based multi-label classification and text-to-text large language models. *Comput. Biol. Med.* **2025**, *196*, 110625. [CrossRef] [PubMed]
20. Artsi, Y.; Klang, E.; Collins, J.D.; Glicksberg, B.S.; Nadkarni, G.N.; Korfiatis, P.; Sorin, V. Large language models in radiology reporting—A systematic review of performance, limitations, and clinical implications. *Intell.-Based Med.* **2025**, *12*, 100287. [CrossRef]
21. Busch, F.; Hoffmann, L.; dos Santos, D.P.; Makowski, M.R.; Saba, L.; Prucker, P.; Hadamitzky, M.; Navab, N.; Kather, J.N.; Truhn, D.; et al. Large language models for structured reporting in radiology: Past, present, and future. *Eur. Radiol.* **2025**, *35*, 2589–2602. [CrossRef] [PubMed]
22. Sanjay Kumar, K.J.; Amritha Nandini, K.L.; Dharshan, S.P.S.; Sowmya, V.; Bandaragoda, T. Automated Crack Analysis and Reporting in Civil Infrastructure using Generative AI. In *IECON 2024—50th Annual Conference of the IEEE Industrial Electronics Society*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–6. [CrossRef]
23. Pu, H.; Yang, X.; Li, J.; Guo, R. AutoRepo: A general framework for multimodal LLM-based automated construction reporting. *Expert Syst. Appl.* **2024**, *255*, 124601. [CrossRef]
24. Elmousalami, H.; Maxy, M.; Hui, F.K.P.; Aye, L. AI in automated sustainable construction engineering management. *Autom. Constr.* **2025**, *175*, 106202. [CrossRef]
25. Lo Duca, A. Using generative AI to co-design data-driven stories. *J. Assoc. Inf. Sci. Technol.* **2025**, *76*, 1786–1802. [CrossRef]
26. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*; NIPS '20; Curran Associates Inc.: Red Hook, NY, USA, 2020.