

# LLMs for Air Quality

Structured Reasoning Meets Zero-Shot Imagination

Authors: **Prof. Nipun Batra**, Abhay Pisharodi, Diya Thakor, Prince Kumar, Ratnesh Pasi, Vedant Acharya



#### Air Pollution in India

- In India alone, more than **1.6 million deaths** each year are attributed to air pollution <sup>[1]</sup>.
- Leading causes: stroke, heart disease, lung cancer, chronic respiratory illness <sup>[1]</sup>
- **PM**<sub>2.5</sub> shortens an average Indian's life expectancy by **5.3** years<sup>[2]</sup>.
- Delhi faces highest impact: 11.9 years shorter lifespan <sup>[2]</sup>.



#### Talk Agenda

#### 1. AQBench

- Benchmarking code-gen abilities of LLMs on Structured Air
   Quality
   Data.
- 2. Open Air LLM
  - Testing LLMs' zero-shot PM<sub>2.5</sub> prediction using natural language prompts.



# **AQBench**

A Benchmark for Structured and Reasoning-Centric Air Quality Querying with LLMs







5



# Can Google search and LLM Chatbots help to answer this question?



6



#### **Google search Response**



Which city in Punjab had the highest PM2.5 concentration in 2024?

In 2024, **Mullanpur**, located in Punjab's Ludhiana district, recorded the highest PM2.5 concentration among cities in the state. According to the IQAir World Air Quality Report 2024, Mullanpur had an average PM2.5 level of **102.3 µg/m<sup>3</sup>**, ranking it as the **third most polluted city in India** and **fourth globally**. True Scoop +1

Other Punjab cities with significant PM2.5 levels included **Mandi Gobindgarh** and **Ludhiana**, which ranked 10th and 14th respectively in India. The Times of India +1

These high pollution levels are primarily attributed to industrial emissions, vehicular pollution, and agricultural stubble burning, especially during the winter months. Reddit +1

Given the health risks associated with elevated PM2.5 levels, using an air purifier can be beneficial. Here are some options:



#### **ChatGPT Response**



#### VA) Which city in Punjab had the highest PM2.5 concentration in 2024?

I'll search for recent information about PM2.5 pollution levels in Punjab cities during 2024.

Punjab cities highest PM2.5 pollution 2024

10 results 🗸

Let me search more specifically for Punjab cities and their PM2.5 levels in 2024.

Mullanpur Punjab PM2.5 2024 most polluted city

10 results 🗸

Based on the search results, Mullanpur in Punjab had the highest PM2.5 concentration among Punjab cities in 2024, with an average PM2.5 concentration of 102.3 µg/m<sup>3</sup> (Oizom) (Down To Earth). Mullanpur was ranked among the top most polluted cities globally, appearing in the world's top 20 most polluted cities list (CNN) (Down To Earth).

The data shows that Mullanpur was ranked the fourth most polluted city in the world in 2024 Top 10 Most Polluted Cities..., making it not just the most polluted city in Punjab but one of the most polluted cities globally. This extremely high PM2.5 concentration is more than 10 times the WHO recommended safe limit of 5 µg/m<sup>3</sup>.



#### **Claude Response**



#### Current dashboards







### Current dashboards







### Current dashboards

#### Most polluted locations near Gandhinagar



**City ranking** Real-time India AQI<sup>+</sup> city ranking # Cities AOI<sup>+</sup> US Delhi, Delhi 1 Noida, Uttar Pradesh 2 💶 Ghaziabad, Uttar Pradesh 3 New Delhi, Delhi 4 5 LIA, West Bengal 6 Lucknow, Uttar Pradesh The Madhyamgram, West Bengal 7 💶 Dam Dam, West Bengal 149 8 147 📥 Jaipur, Rajasthan 9 143 Kanpur, Uttar Pradesh 10



Worldwide AQI\* ranking -



# Correct answer based on the 2024's CPCB official data is Mandi Gobindgarh.



#### How did we get the correct answer?





#### Step-1: Downloading CPCB data



Data-Repo Link: https://airquality.cpcb.gov.in/ccr/#/caaqm-dashboard-all/caaqm-landing/caaqm-data-repository



#### Step-1: Downloading CPCB data

-	Central Control Room for	Air Quality Manage	ment - All India	) Su	CCESS a fetched successfully!
Data Repositor	у				
Data type	Raw data 🛛 🗙 🔻	Frequency	1 day	× •	
State	Punjab X	City	Amritsar	× •	
Station	Golden Temple, Amritsar - PPCB 🛛 🗙 🔻			Submit	
Data Files					
	Data Files (year wise)			Downloads	Click on downloa
	2024				button against th
	2023			*	required year
	2022			2	
	2021			<u>خ</u>	
	2020			*	
2019				*	
	2018			2	

Data-Repo Link: https://airquality.cpcb.gov.in/ccr/#/caaqm-dashboard-all/caaqm-landing/caaqm-data-repository

#### Step-1: Downloading CPCB data

• The dataset for each station will look like this

Timestamp	PM2.5 (μg/m³)	PM10 (μg/m³)	NO (µg/m³)	NO2 (µg/m³)	NOx (ppb)	NH3 (µg/m³)	SO2 (µg/m³)	CO (mg/m³)
1/1/2024	74.39	165.98	13.65	18.17	29.36	34.23	6.52	0.17
1/2/2024	70.17	155.41	16.26	20.05	38.01	34.77	6.63	0.19
1/3/2024	68.43	151.06	20.45	20.17	36.6	30.19	9.74	0.30
1/4/2024	63.97	139.92	28.66	20.88	47.25	25.47	13.20	0.45
1/5/2024	70.38	155.94	28.55	20.67	48.28	25.10	14.07	0.48
1/6/2024	59.17	127.92	28.1	21.73	47.71	24.66	12.72	0.43
1/7/2024	60.84	132.11	31.37	16.93	45.67	17.78	12.71	0.43
1/8/2024	80.75	181.88	37.37	18.64	52.94	18.32	13.36	0.45

# Step-2 Writing a Python Code

- All the station-wise csvs are concatenated in a single dataframe.
- 'city' & 'station' columns are also added in the dataframe.

#### Python script

city\_df = punjab\_df.groupby('city')['PM2.5 (µg/m<sup>3</sup>)'].mean().reset\_index()
correct\_ans = city\_df.sort\_values('PM2.5 (µg/m<sup>3</sup>)').iloc[-1]['city']
print(correct\_ans)



### Introduction to AQBench

- Contains 10,000+ air pollution-related queries paired with ground-truth Python code.
- Questions related to "Funding for better Air Quality" and "State demography" are included as well.
- Built on curated tabular data including:
  - CPCB data
  - NCAP funding data
  - State Demographic data



#### Central Pollution Control Board (CPCB)

- CPCB aims to monitor and reduce water pollution and air pollution in India.
- Installed 500+ air quality sensors across india, under National Air Monitoring Programme (NAMP).
- Sensors provide datasets offering rich, long-term insights.



### CPCB dataset

• Added station, city, latitude, and longitude columns after merging all station-wise CSVs from 2017–2024.

Timestamp	station	PM2.5	PM10	address	city	latitude	longitude	state
1/1/2017	Ardhali Bazar, Varanasi - UPPCB	166.32	213.93	Ordali Bazar, Tagore Town, Orderly Bazar, Varanasi, Uttar Pradesh 221002	Varanasi	25.3506	82.90831	Uttar Pradesh
1/1/2017	Bandra, Mumbai - MPCB	118.20	270.92	CAAQMS Station Govt Polytechnic Mumbai, Near Gov Holi Tecni, Mumbai.	Mumbai	19.0627	72.84614	Maharashtra
1/1/2017	Gangapur Road, Nashik - MPCB	84.48	104.14	Gangapur Road, Nashik, Maharashtra 422002	Nashik	20.00733	73.77624	Maharashtra



#### CPCB dataset schema

Column	Data Type	Description
Timestamp	datetime64[ns]	Date on which the measurement was recorded
station	object	Name of the monitoring station
PM2.5	float64	Daily mean of fine particulate matter (PM2.5 in $\mu$ g/m <sup>3</sup> )
PM10	float64	Daily mean of coarse particulate matter (PM10 in $\mu$ g/m <sup>3</sup> )
address	object	Physical address of the monitoring station
city	object	City where the monitoring station is located
latitude	float64	Latitude coordinate of the monitoring station
longitude	float64	Longitude coordinate of the monitoring station
state	object	State or Union Territory of the city



#### National Clean Air Programme (NCAP)

- Objective: Improve air quality in 131 non-attainment and million-plus cities across 24 States/UTs
- Allocation: Over ₹9,650 crore released to 131 cities from FY 2019-20 to FY 2023-24.
- Utilization: Funds support air quality monitoring networks, capacity building, and research



## NCAP funding dataset

S. No.	state	city	Amount released during FY 2019-20	Amount released during FY 2020-21	Amount released during FY 2021-22	Total fund released	Utilisation as on June 2022
1	Andhra Pradesh	Guntur	0.12	0.76	1.96	2.84	22.91
2	Andhra Pradesh	Kurnool	0.06	0.76	1.36	2.18	22.91
3	Andhra Pradesh	Nellore	0.06	0.76	1.92	2.74	22.91
4	Bihar	Gaya	0.10	2.00	1.90	4.00	15.20
5	Bihar	Muzaffarpur	0.10	5.00	2.50	7.60	15.20
6	Chandigarh	Chandigarh	8.28	5.00	4.61	17.89	10.83
7	Himachal Pradesh	Baddi	0.06	3.00	0.20	3.26	6.35
8	Himachal Pradesh	Paonta Sahib	0.06	1.00	0.10	1.16	6.35
9	Himachal Pradesh	Sunder Nagar	0.06	1.00	0.08	1.14	6.35
10	Jammu & Kashmir	Jammu	0.12	3.00	4.89	8.01	0.12



#### NCAP Funding Dataset schema

Column	Data Type	Description
S. No.	int64	Serial number as per the original source file
state	object	State or Union Territory of the city
city	object	City that received funding under NCAP
Amount released FY 2019–20	float64	Funds released in FY 2019–2020 (in INR)
Amount released FY 2020-21	float64	Funds released in FY 2020–2021 (in INR)
Amount released FY 2021–22	float64	Funds released in FY 2021–2022 (in INR)
Total fund released	float64	Total funds allocated to the city across the years
Utilisation as on June 2022	float64	Total amount utilized from the fund as of June 2022



### Demographic data

- Demographic data is the state wise population, area and administrative setup data.
- From the **2011 census** data available over internet.
- True/False is marked state wise weather it is Union Territory or not respectively.





### States Demographics dataset schema

Column Data Type		Description
state	object	Name of the Indian state or union territory
population	int64	Total population of the state
area (km²)	int64	Total geographical area of the state in square kilometers
isUnionTerritory	bool	Boolean flag indicating if the region is a Union Territory



## **AQBench Construction**

- 1. Defined 7 categories for questions
- 2. Created question templates and instances
- 3. Sampled question instances
- 4. Paraphrased the question instances



#### Question Categories: # 1 Spatial Aggregation

Group data by location (station, city, state) for a specific time range.

**Example:** Which station in *Delhi* had the highest average PM<sub>2.5</sub> in Jan 2022?



https://aqicn.org/map/delhi/



#### Question Categories: # 2 Temporal Aggregation

Analyze how pollution values change over time at a single or multiple locations.

**Example**: How did PM<sub>2.5</sub> vary across 2021 in Lucknow?



https://huggingface.co/spaces/SustainabilityLabIITGN/VayuBuddy



# Question Categories: # 3 Spatio-Temporal Aggregation

Combine time and geography in analysis (e.g., state-wise seasonal comparison).

**Example**: Which state had the highest PM<sub>2.5</sub> in summer 2022?



https://www.iqair.com/in-en/air-quality-map?zoomLevel=4&lat=22.88578 21183&lng=79.6119761026



#### Question Categories: # 4 Specific Patterns

Focus on specific date ranges (not overall trends).

# **Example**: How many days in the last 5 years did Mumbai violate PM<sub>2.5</sub> limits?



# exposure.

**Example**: Which state has the fewest stations relative to its population?

population data to evaluate

## Question Categories: # 5 Population-based Aggregation Combine air quality and





#### Question Categories: # 6 Area-Based Aggregation

Use land area (not region name) for grouping and comparison.

**Example**: Which state over 50,000 km<sup>2</sup> had the lowest PM<sub>10</sub> level?



JSTAINABILITY

#### Question Categories: # 7 Funding Related

Analyze NCAP fund allocation and usage for policymaking.

**Example**: Which state with NCAP funding has the lowest PM<sub>2.5</sub> levels?

state	city	Amount released during FY 2019-20	Amount released during FY 2020-21	Amount released during FY 2021-22
Andhra Pradesh	Guntur	0.12	0.76	1.96
Andhra Pradesh	Kurnool	0.06	0.76	1.36
Andhra Pradesh	Nellore	0.06	0.76	1.92
Bihar	Gaya	0.10	2.00	1.90
Bihar	Muzaffarpur	0.10	5.00	2.50



### Question Templates v/s Instances

#### Template

Which **{location}** has **{stats} {method} {pollutant}** in year **{year}**?

#### Instances

Which Station has highest mean PM<sub>2.5</sub> in year 2021 ?
Which City has 2<sup>nd</sup> highest median PM<sub>2.5</sub> in year 2022 ?
Which State has lowest mean PM<sub>10</sub> in year 2023 ?
Which Station has 2<sup>nd</sup> lowest median PM<sub>10</sub> in year 2024 ?


#### **Code Template and Instance**

Code Template in JSON object and Code Instances in Python Script for the below question. **Question** : How many times Mumbai city crossed the  $60\mu$ g/m<sup>3</sup> of PM<sub>2.5</sub> in year 2021 ?

#### Code Template

```
def true_code():
    import numpy as np
    import pandas as pd
    data = pd.read_pickle("preprocessed/data.pkl")
    data = data[data['Timestamp'].dt.year == {year}]
    data = data[data['city'] == "{city}"]
    data = data[data['city'] == "{city}"]
    data = data.dropna(subset="{col}")
    data = data[data["{col}"] > {limit}]
    count = data['Timestamp'].nunique()
    print(count)
true_code()
```

#### Code Instance

```
def true_code():
    import numpy as np
    import pandas as pd
    data = pd.read_pickle("preprocessed/data.pkl")
    data = data[data['Timestamp'].dt.year == 2021]
    data = data[data['city'] == "Mumbai"]
    data = data[data['city'] == "Mumbai"]
    data = data[data['PM2.5"] > 60]
    count = data['Timestamp'].nunique()
    print(count)
true_code()
```



## **Sampling** Questions

#### **Goal:** Reduce redundancy, lower compute utilization

- Generated **27,000+** questions using templates
- Many had **repetitive patterns** (e.g., stations, dates)
- Sampled **10,034 diverse questions** across templates
- Balanced template coverage and efficiency



## **Paraphrasing** Questions

#### **Goal: Increase linguistic variety**

- Template-based questions were structurally similar
- Used Gemini 2.5 Pro to paraphrase for natural variation
- Ensured placeholders (columns, methods) stayed correct
- Manual checks maintained alignment with expected code



#### Example of Paraphrased Questions

<b>Original Questions</b>	Paraphrased Questions
Which station has the 2nd lowest average PM <sub>10</sub> in July 2022 ?	Determine the station with the second-lowest average PM <sub>10</sub> reading for July 2022.
Which weekday (i.e. Monday, Tuesday, Wednesday etc) sees the 2 <sup>nd</sup> highest PM <sub>2.5</sub> pollution levels on average in 2022?	In 2022, on which day of the week were average PM <sub>2.5</sub> pollution levels the second highest?
Which date in the last five years recorded the lowest $PM_{2.5}$ in the Jaipur?	In Jaipur, which date during the last five years had the lowest recorded PM <sub>2.5</sub> level?



#### **Benchmark Questions Composition**

Category	Templates	Instances	Selected Instances
SA Spatial Aggregation	12	14288	4897
TT Temporal Trends	9	2988	1219
STA Spatio-Temporal Aggregation	7	6046	2463
SP Specific Patterns	5	624	256
PB Population Based Exposure	12	926	383
AB Area Based Aggregation	10	900	373
FQ Funding Related Questions	11	1070	443





#### **Our System Prompt**

```
' station ': dtype ('< M8 [ HS ]')
' station ': dtype ( '0') ,
' PM2 .5 ': dtype ( 'float64') ,
' PM10 ': dtype ( 'float64') ,
' address ': dtype ( '0') ,
' city ': dtype ( '0 ') ,
' latitude ': dtype ( ' float64 ') ,
' longitude ': dtype ( ' float64 ') ,
' state ': dtype ( '0 ')</pre>
```



42

}

#### **Our System Prompt**

```
2. ' states_data ', a pandas DataFrame of state - wise population , area and whether state
is union territory or not of India . The states_data has the following columns and data
types :
{
      ' state ': dtype ( '0 ') ,
      ' population ': dtype ( ' int64 ') ,
      ' area ( km2 ) ': dtype ( ' int64 ') ,
      ' isUnionTerritory ': dtype ( ' bool ')
}
3. ' ncap_funding_data ' , a pandas DataFrame of funding given to the cities of India from
2019 -2022 , under The National Clean Air Program ( NCAP ) .
{
    'S . No . ': dtype ( ' int64 ') ,
     ' state ': dtype ( '0 ') ,
     ' city ': dtype ( '0 ') ,
     ' Amount released during FY 2019 -20 ': dtype ( ' float64 ') ,
     ' Amount released during FY 2020 -21 ': dtype ( ' float64 ') ,
     ' Amount released during FY 2021 -22 ': dtype ( ' float64 ') ,
     ' Total fund released ': dtype ( ' float64 ') ,
     ' Utilisation as on June 2022 ': dtype ( ' float64 ')
}
```



#### **Our System Prompt**

```
You just have to generate the code , don't give any explanation . The code should
always return a single value not a rows , columns , Data Frames or tuples . Do Not
change the function name it will be always ' get_response ' as in function signature
. Always import necessary libraries . Write the code in <code>.
Function signature :
def get_response ( data : pd . DataFrame , states_data : pd . DataFrame ,
ncap_funding_data : pd . DataFrame ) :
# Your code here
Example :
>>> What 's the maximum PM2 .5 recorded ever ?
<code>
import pandas as pd
import numpy as np
def get_response ( data : pd . DataFrame , states_data : pd . DataFrame ,
ncap_funding_data : pd . DataFrame ) :
max_pm25 = data [ ' PM2 .5 ']. max ()
return max_pm25
</code>
```



#### Flow of AQBench Construction





#### Steps for LLM Evaluation on AQBench



08-06-2025



08-06-2025

#### **Evaluation Metrics**

• **exec@1** : Measures the proportion of generated code that runs runs without errors, regardless of correctness, reflects syntactic and execution reliability.

exec@1 =	Number of samples that run without errors
	Total number of samples

• **pass@k (n=5)**: Fraction of attempts where at least one of the top-k outputs passes all test cases, shows model's ability to generate correct code within n tries.

$$pass@k = \frac{1}{N} \sum_{i=0}^{N} 1 - \frac{\binom{n-c}{k}_{i}}{\binom{n}{k}_{i}}$$



#### **Evaluation Metrics**

#### **Syntactic Error Definition**

- Occurs when the generated code contains syntax errors
- Causing the Python interpreter to raise a compilation error.

**Error rate:** Defined as the proportion of code instances in which a syntactic error occurred.

Error Rate =	Number of syntactic error codes
	Total number of codes generated



#### Models Used

#### Code-gen LLMs:

- Qwen-2.5-Coder-1.5B
- Qwen-2.5-Coder-3B
- Qwen-2.5-Coder-7B
- Qwen-2.5-Coder-14B
- Qwen-3-1.7B
- Qwen-3-32B

#### **General-purpose LLMs:**

- LLaMA-3.2-1B
- LLaMA-3.2-3B
- Mistral-7B-instruct-v0.3



#### LLMs' Performance on AQBench



- Code-gen Qwen2.5-Coder-3B performs better than general-purpose LLaMA-3.2-3B of the same size.



08-06-2025

#### LLMs' Performance on AQBench

Model Name	exec@1	pass@1	pass@2	Error rate
Llama3.2-1B	0.04	0.01	0.01	0.96
Llama3.2-3B	0.17	0.03	0.07	<u>0.83</u>
Mistral-7B-instruct-v0.3	0.22	0.02	0.04	0.77
Qwen-2.5-Coder-1.5B	0.47	0.07	0.11	0.52
Qwen-2.5-Coder-3B	0.73	0.31	0.44	0.26
Qwen-2.5-Coder-7B	0.78	0.38	0.49	0.21
Qwen-2.5-Coder-14B	<u>0.94</u>	<u>0.64</u>	<u>0.69</u>	0.05
Qwen-3-1.7B	0.46	0.05	0.07	0.53
Qwen-3-32B	0.98	0.72	0.75	0.02

## Qwen3-32B outperforms all, but Qwen2.5-Coder-14B shows comparable results (4% lower exec@1, 8% lower pass@1)



#### Category-wise pass@1 performance



08-06-2025

#### Summary of results

- Qwen2.5-Coder-3B performs better than general-purpose LLaMA-3.2-3B of the same size.
- Higher-parameter models perform better overall. Qwen3-32B outperforms all, but Qwen2.5-Coder-14B shows comparable results (4% lower exec@1, 8% lower pass@1).
- Qwen2.5-Coder-14B also beats Qwen3-32B in SP (pass@1) despite fewer parameters.



**Syntactic error:** Occurs when the generated code contains syntax errors, causing the Python interpreter to raise a compilation error.

#### Sub-Categories in syntactic error:

- 1. SyntaxError
- 2. ColumnSelectionError
- 3. NameError
- 4. ImportError
- 5. OtherError



- **1. SyntaxError**: Invalid Python syntax (e.g., bad indentation, pm2.5\_state as var) Example:
  - **Question:** Determine the city that recorded the third greatest average PM<sub>2.5</sub> ever.

```
import pandas as pd
def get_response(data: pd.DataFrame, states_data: pd.DataFrame, ncap_funding_data: pd.DataFrame):
    city_pm25 = data.groupby('city')['PM2.5'].mean()
    sorted_city_pm25 = city_pm2.5.sort_values(ascending=False)
    third_greatest = sorted_city_pm25.iloc[2]
    return sorted_city_pm25.index[2]
```

 city\_pm2.5 is used in variable access which is not a valid variable name, causing SyntaxError



**2. ColumnSelectionError:** Occurs when model refers to invalid/misspelled columns/indices.

Example:

- **Question:** Identify the station that recorded the minimum PM<sub>2.5</sub> level on January 27, 2019.

```
import pandas as pd
import numpy as np
def get_response(data: pd.DataFrame, states_data: pd.DataFrame, ncap_funding_data: pd.DataFrame):
    filtered_data = data[data['Timestamp'].dt.strftime('%Y-%m-%d') == '2019-01-27']
    min_pm25_station = filtered_data.loc[data['PM2.5'].idxmin()]['station']
    return min_pm25_station
```

- Trying to access to index data['PM2.5'].idxmax() on a filtered DataFrame, which isn't present in filtered\_data, leads to KeyError.
- Should use filtered\_data['PM2.5'].idxmax() instead.



3. NameError: Occurs when model uses undefined variables/functions.

Example:

- **Question:** Name the station that registered the minimum average PM<sub>2.5</sub> level in August 2018.

imp	ort pandas as pd
def	<pre>get_station():</pre>
	<pre>data['Timestamp'] = pd.to_datetime(data['Timestamp'])</pre>
	<pre>august_2018 = data[(data['Timestamp'].dt.month == 8) &amp; (data['Timestamp'].dt.year == 2018)]</pre>
	<pre>min_avg_pm25 = august_2018['PM2.5'].mean()</pre>
	<pre>min_station = august_2018.loc[ august_2018['PM2.5'].idxmin(),'station']</pre>
	return min_station

- Didn't use the **get\_response(data,states\_data, ncap\_funding\_data)** function that we asked to write in system\_prompt, so **data** is not defined



**4. ImportError**: Importing modules/loading files which are not present.

Example:

- **Question:** Identify the station that recorded the minimum PM<sub>2.5</sub> level on January 27, 2019.



- pandas don't have any module named pd



**5. OtherError:** Misc. issues like merge errors, type mismatches, or control flow misuse. This type of errors are very less.

Question: Report the station with the 3rd highest average PM<sub>2.5</sub> in September 2024



- Can only pass argument "on" OR "left\_on" and "right\_on", not a combination of both.







08-06-2025

#### Summary of Syntactic Error Analysis

- Smaller models (e.g., **LLaMA-3.2-1B**) show weaker Python syntax understanding, leading to more errors.
- Larger models have fewer errors overall; **ColumnSelectionError** is the most common type.
- Qwen3-32B, Qwen2.5-Coder-14B, and 7B all have syntactic error rates under 0.1%.
- Code-generation models (Qwen2.5-Coder, Qwen-3) indicates stronger understanding of Python syntax.
- Mistral-7B has the highest ColumnSelectionError rate despite its size.



#### Impact

- Empowers **policy analysts**, **NGOs**, and **researchers** to draw insights from air quality data without coding.
- Breaks technical barriers by enabling natural language access to structured datasets.
- Accelerates research and advocacy through intuitive data querying.
- Bridges the gap between **public data** and **public understanding** using **LLMs**.



#### Future Vision for AQBench

- **Satellite integration** to detect pollution hotspots using NASA/ESA data.
- Real-time AQI answers via CPCB and OpenAQ API streams.
- Deploy **tiny LLMs on edge devices** for fast, offline, private inference.
- Enable **Multimodal reasoning** over text, tables, and geographic data.



#### Learning from AQBench

#### What We Learned from Benchmarking LLMs ?

- LLMs can reason over structured data, but performance varies.
- Queries involving joins, aggregations, or temporal filters are harder.
- Prompt engineering (Our System Prompt) and example-driven methods improve outcomes.
- Evaluation with **pass@k** gives realistic measure of reliability.



# What If There's No Structured Data at All?

- AQBench shows LLMs can reason with structured data and clear schemas
- But what happens when we strip all that away?
- Can LLMs make educated guesses about air quality using just their training knowledge?
- What assumptions do LLMs make about air quality?



## **OpenAirLLM**

# Exploring zero-shot LLM predictions of PM2.5 and uncovering hidden biases



## Zero-Shot LLMs for predicting PM<sub>2.5</sub>

**Key Question:** Can **ChatGPT or other LLMs** Predict Your City's Air Pollution?

- LLMs like ChatGPT are now asked about everything including environment information
- But these models aren't built for environmental reasoning



## Zero-Shot LLMs for predicting PM<sub>2.5</sub>

- Actual values from CPCB (Central Pollution Control Board)
  - Begusarai: 148.31 μg/m<sup>3</sup>
     Delhi: 102.18 μg/m<sup>3</sup>
- Begusarai underpredicted
- Delhi overpredicted
- Why such **inconsistencies**?

Estimate the average PM2.5 concentration in ( $\mu$ g/m^3) for the following location in the year 2023, City: Begusarai

70

ISTAINABILITY

Only give the PM2.5 value in response

64.8 μg/m^3

Estimate the average PM2.5 concentration in ( $\mu$ g/m^3) for the following location in the year 2023, City: Delhi

Llama-3.3-70B

**?** Only give the PM2.5 value in response

108.1 µg/m^3

#### Motivation

- With growing reliance on LLMs, understanding when they go
   wrong and why is crucial.
- LLMs are often treated as black-box knowledge agents.
- Air pollution is a complex challenge influenced by spatial, seasonal, and socioeconomic dynamics

#### Goals:

- Assess how well LLMs predict PM<sub>2.5</sub>.
- Identify hidden biases in these predictions.



#### Investigations

- How well do LLMs predict PM<sub>2.5</sub> concentrations at different spatial and temporal granularities?
  - Do LLMs Understand the Plausible Ranges of PM<sub>2.5</sub> Values?
  - Can LLMs Predict PM<sub>2.5</sub> Concentrations Across diverse Indian Cities?
  - Do LLMs Predictions Correlate with Actual PM<sub>25</sub>?
  - Do LLMs have Seasonal Variation in their Predictions (National aggregation)?


### Investigations

- Do LLMs exhibit **geographic and socioeconomic bias** in their predictions?
  - Are highly polluted lower-tier cities underrepresented?
  - Do LLMs assume greener cities are less polluted?
  - Do LLMs overpredict PM<sub>25</sub> for well-known cities?
  - Does city name visibility affect PM<sub>2.5</sub> predictions from LLMs?
  - Do LLMs underestimate PM<sub>2.5</sub> levels in cities with lower socioeconomic status?



### Investigations

- Do LLMs show sensitivity to prompt design and auxiliary information in their predictions?
  - Does adding more context to prompts reduce prediction error?
  - Does prompt rephrasing lead to different LLM predictions?

## Model Used

- Gemma-2-9B-IT
- Llama-3.1-8B-Instruct
- Gemma-3-27B-IT
- Llama-3.3-70B-Instruct

These models are most downloaded models in Hugging Face



#### Dataset

#### Ground Truth Data: CPCB (Central Pollution Control Board)

- Station-level PM<sub>2.5</sub> data across India
- Multiple stations per city

#### How We Aggregated Ground Truth as per prompt variation:

- City-level (Yearly):
  - Average of all stations in a city over the full year
- City-level (Monthly):
  - Average of all stations in a city for each month
- National-level (Monthly):
  - Average of all stations across India for each month



#### **Evaluation procedure**

Mean Absolute Error:

Underprediction %:

Overprediction %:

$$egin{split} rac{1}{C}\sum_{c=1}^C \left(rac{1}{N_c}\sum_{i=1}^{N_c}\left|\hat{y}_i^{(c)}-y_{ ext{true}}^{(c)}
ight|\ &rac{1}{C}\sum_{c=1}^C\mathbb{1}\left(ar{\hat{y}}_c < y_{ ext{true},c}
ight) imes 100\%\ &rac{1}{C}\sum_{c=1}^C\mathbb{1}\left(ar{\hat{y}}_c > y_{ ext{true},c}
ight) imes 100\% \end{split}$$



### **Prompt Used**

#### • Prompt 1:

Estimate the average  $PM_{2.5}$  concentration (in  $\mu g/m^3)$  for the following location in the **year 2023** City: {city}, {state}

#### • Prompt 2:

What is the average  $\text{PM}_{2.5}$  (in  $\mu\text{g/m}^3)$  value in month {month} in year 2023 in India?

#### • Prompt 3:

Provide average  $PM_{2.5}$  (in  $\mu g/m^3$ ) pollution values for the **year 2023** in **month of {month}** for the below location City: {city}, State: {state}



# Do LLMs understand the plausible ranges of $PM_{2.5}$ Values?

Model	Below 0 µg/m³ (%)	Above 500 μg/m³ (%)
gemma2-9b	0	0
Llama-3.1-8B	0	0
gemma-3-27b	0	0
Llama-3.3-70B	0	0

Proportion of predictions outside the plausible  $PM_{2.5}$  range (< 0 or > 500µg/m<sup>3</sup>) for each model. Evaluated on Prompt 1, 2 and 3



# Can LLMs predict PM<sub>2.5</sub> concentrations across diverse Indian cities?

 When tested across all CPCB cities, LLMs can produce plausible PM<sub>2.5</sub> values – but show varying degrees of overand underprediction.

Model	MAE	Overpredicted (%)	Underpredicted (%)
Gemma2-9b	24.48	6.81	93.19
Llama3.1-8b	18.66	27.23	72.77
Gemma3-27b	14.43	52.34	47.66
Llama3.3-70b	20.65	74.47	25.53



# Can LLMs predict PM<sub>2.5</sub> concentrations across diverse Indian cities?

- Smaller models are mostly doing underprediction but still it exhibit biases for city **Delhi** relative to other cities Prediction.
- Bigger models are clearly showing underprediction and overprediction.



Top 10 Most Polluted Indian Cities in 2023



# Do LLMs predictions correlate with actual PM<sub>2.5</sub>?

- Larger LLMs better capture real-world pollution patterns.
- Their predictions show stronger correlation with actual PM<sub>2.5</sub>



# Do LLMs have seasonal variation in their predictions?

- Llama-3.3-70B-Instruct shows clear seasonal understanding
- All other smaller models produce nearly flat predictions failing to capture winter peaks and summer lows.

Monthly PM2.5 — Ground Truth vs Prediction (2023) National Level



### Are highly polluted tier-3 cities underpredicted?

Model	Underprediction(%)	MAE
gemma2-9b	100.00	67.08
Llama-3.1-8B-Instruct	100.00	56.83
gemma-3-27b	83.33	40.03
Llama-3.3-70B-Instruct	80.00	26.76

#### 30 most polluted tier-3 cities in 2023

• Most-polluted Tier-3 cities include Byrnihat, Begusarai, Saharsa, Katihar, Purnia, etc.



84

#### Are highly polluted tier-3 cities underpredicted?



30 most polluted tier-3 cities in 2023

- Smaller models underpredicted PM<sub>2.5</sub> for all 30 highly polluted Tier-3 cities.
- Even larger models showed significant underprediction.



# Do LLMs assume greener cities are less polluted?

- To look for this bias we used NDVI (Normalized Difference Vegetation Index)
- It measures how green or vegetated a place is using satellite data.
- High NDVI = dense greenery

08-06-2025

• Low NDVI = dry, barren, or urban land



# Do LLMs assume greener cities are less polluted?

 This suggests that there is high correlation between NDVI and PM<sub>2.5</sub> prediction



# Do LLMs assume greener cities are less polluted?

 But there seems not any much correlation between actual PM<sub>2.5</sub> and Average NDVI which shows biases



## Do LLMs overpredict PM<sub>2.5</sub> for well-known cities?

Model	MAE	Overpredict (%)
Gemma2-9b	21.4	0.0
Llama3.1-8b	16.8	17.8
Gemma3-27b	8.4	57.1
Llama3.3-70b	14.2	75.0

Results for Top 30 most searched cities from google search count

- Most-searched cities include Delhi, Mumbai, Bengaluru, Chandigarh, etc.
- From table we can say that overpredicted % is higher in case of well-known cities



## Do LLMs overpredict PM<sub>2.5</sub> for well-known cities?



Results from Top 30 most searched cities from google search count



## Does city name visibility affect PM<sub>2.5</sub> predictions from LLMs?

city	state	news_count
delhi	delhi	5285
gurugram	haryana	849
noida	uttar pradesh	368
patna	bihar	317
ghaziabad	uttar pradesh	86
faridabad	haryana	51
meerut	uttar pradesh	44

Top-7 most coverage cities by TOI and The Hindu cities from 2010-21 (PM2.5 >70 $\mu$ g/m<sup>3</sup>) From **Samachar** Paper



91

## Does city name visibility affect PM<sub>2.5</sub> predictions from LLMs?



Highly visible cities (article count > 1) with  $PM_{25} > 70 \ \mu g/m^3$ 



## Does city name visibility affect PM<sub>2.5</sub> predictions from LLMs?



Low-visibility cities (article count  $\leq$  1) with PM<sub>2.5</sub> > 70 µg/m<sup>3</sup>

• Cities with **higher visibility** (more articles) tend to have **higher PM**<sub>2.5</sub> over predictions, while lesser-known cities face more underprediction

08-06-2025

SUSTAINABILITY

# Do LLMs underestimate PM<sub>2.5</sub> levels in cities with lower socioeconomic status?



20-most Poorer and highly polluted cities (per capita income < 100,000 INR)

• PM<sub>2.5</sub> is often **underpredicted** in **lower-income cities**, revealing a bias tied to socioeconomic status.



94

#### **State-Level Bias Patterns**



• Bias patterns (under/over prediction) vary significantly by state and model scale

08-06-2025

**SUSTAINABILITY** 

# Does adding more context to prompts reduce prediction error?

• P1(City Only)

Estimate the average  $PM_{2.5}$  concentration (in  $\mu g/m^3$ ) for the following location in year 2023. City: {city}

• P2(City + State)

Estimate the average  $PM_{2.5}$  concentration (in  $\mu g/m^3$ ) for the following location in year 2023. **City: {city}, {state}** 

• P3(City + State + Nearby Information)

Estimate the average  $PM_{2.5}$  concentration (in  $\mu g/m^3$ ) for the following location in year 2023. City: {city}, {state} Nearby: {city1 (X km)}, {city2 (Y km)}



# Does adding more context to prompts reduce prediction error?

Model	MAE(P1 / P2 / P3)	Underpredict % (P1 / P2 / P3)
Gemma2-9b	23.2 / 24.4 / 19.9	91.4 / 93.1 / 86.3
Llama3.1-8b	19.0 / 18.6 / 19.7	71.9 / 72.7 / 77.4
Gemma3-27b	15.0 / 14.4 / 16.3	49.7 / 47.6 / 34.8
Llama3.3-70b	17.3 / 20.6 / 20.4	46.3 / 25.5 / 26.8

- Adding context does not consistently reduce prediction error across models.
- Smaller models (e.g., Gemma2-9b) benefit slightly from additional context, while larger models show mixed or inverse trends.
- Underprediction rates vary more clearly added context tends to reduce underprediction in larger models



# Does prompt rephrasing lead to different predictions?

• P1:

Estimate the average  $PM_{2.5}$  concentration (in  $\mu g/m^3$ ) for the following location in year 2023. City: {city}, {state}

#### • P2:

What was the average  $PM_{2.5}$  concentration (in  $\mu g/m^3$ ) in the year 2023 for the city of **{city}, {state}?** 

• P3:

Average  $PM_{2.5}$  (in µg/m<sup>3</sup>) in 2023 for {city},{state}?

# Does prompt rephrasing lead to different LLM predictions?

Model	MAE(P1 / P2 / P3)	Under-Pct (P1 / P2 / P3)
Gemma2-9b	24.4 / 22.8 / 17.1	93.1 / 90.6 / 72.3
Llama3.1-8b	18.6 / 23.1 / 20.3	72.7 / 85.9 / 79.1
Gemma3-27b	14.4 / 16.0 / 15.6	47.6 / 45.9 / 37.0
Llama3.3-70b	20.6 / 17.0 / 18.8	25.5/ 64.2/35.7

- **Prompt rephrasing** causes **large variations** in prediction error and bias, especially in **smaller models**.
- Larger models (e.g., Llama3.3-70B) are more stable but still sensitive to prompt design.



99

## Summary

- Larger LLMs show stronger correlation with actual PM<sub>2.5</sub> values and capture seasonal trends better.
- Tier-3 cities are mostly underpredicted.
- High NDVI–PM<sub>2.5</sub> correlation suggests models assume greener cities have cleaner air.
- Well-known cities are overpredicted and lower-income and low visibility cities are underpredicted



#### Takeaways: Structured Vs Zero-Shot

#### AQBench

- Delivers accurate and consistent answers using structured prompts.
- Outputs are easier to trace and interpret.
- Works well for analytical and policy-related queries.

#### **OpenAirLLM**

- Generates creative responses but lacks reliability.
- Predictions vary with slight prompt changes.
- Struggles with **biases** and complex reasoning without guidance.



#### What we need?

- **Hybrid systems** that blend prompts, structure, and grounding
- Structured inputs reduce ambiguity and prompts enhance usability
- Grounding in real data ensures relevance and trust
- Future tools should support both flexibility and rigor



#### **Broader Lessons**

- LLMs are not just search engines they must reason over data
- Prompt sensitivity can limit consistency
- Structure helps guide models through complex queries
- Bias and reliability still need active mitigation



### References

[1] A. Pandey *et al.*, "Health and economic impact of air pollution in the states of India: the Global Burden of Disease Study 2019," *The Lancet Planetary Health*, vol. 5, no. 1, pp. e25–e38, Dec. 2020, doi: 10.1016/s2542-5196(20)30298-9.

[2] University of Chicago, "Potential gain in life expectancy from permanently reducing PM2.5 from 2021 concentration to the WHO guideline in the 10 most populous states of India," report, 2021. [Online]. Available: <u>https://aqli.epic.uchicago.edu/wp-content/uploads/2023/08/India-FactSheet-2023\_Final.pdf</u>



## Thank you



Prof. Nipun Batra Associate Professor, IIT Gandhinagar https://nipunbatra.github.io/

Lab Website: https://sustainability-lab.github.io/



### Prerequisites for using AQBench

• How LLMs work ?

They are trained on massive datasets and learn to **predict the next word** in a sequence based on the preceding words, enabling them to perform tasks like **text generation**, translation, and **answering questions**.

• What is **System Prompt**?

A set of instructions that define how an model should behave across all interactions with users.

