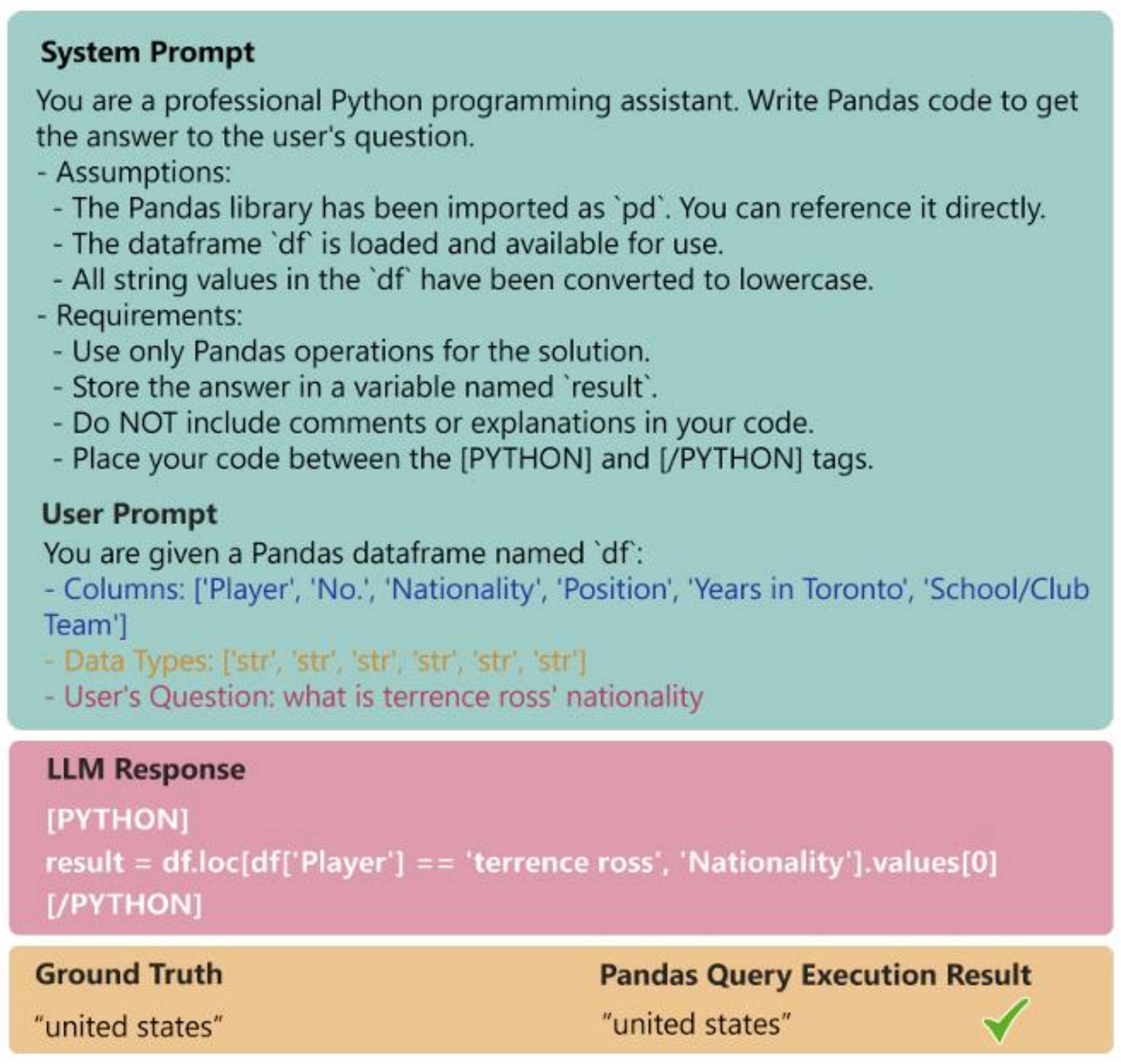
DataFrame QA: A Universal LLM Framework on DataFrame Question Answering Without Data Exposure

Junyi Ye, Mengnan Du, Guiling Wang
New Jersey Institute of Technology, USA

Introduction

Motivation:

- This paper introduces **DataFrame QA**, a novel task that utilizes **LLMs** to generate Pandas queries for information retrieval and data analysis on dataframes, emphasizing safe and non-revealing data handling.
- Emphasizes **data privacy** by relying on dataframe column names and types, not values.
- Addresses limitations of existing LLM-based data analysis methods, such as:
  - **Token cost** in querying large tables.
  - Risk of **data leakage**.
  - Challenges with **mathematical calculations/data analysis**.



System Prompt

You are a professional Python programming assistant. Write Pandas code to get the answer to the user's question.
- Assumptions:
  - The Pandas library has been imported as `pd`. You can reference it directly.
  - The dataframe `df` is loaded and available for use.
  - All string values in the `df` have been converted to lowercase.
- Requirements:
  - Use only Pandas operations for the solution.
  - Store the answer in a variable named `result`.
  - Do NOT include comments or explanations in your code.
  - Place your code between the [PYTHON] and [/PYTHON] tags.

User Prompt

You are given a Pandas dataframe named `df`:
- Columns: ['Player', 'No.', 'Nationality', 'Position', 'Years in Toronto', 'School/Club Team']
- Data Types: ['str', 'str', 'str', 'str', 'str', 'str']
- User's Question: what is terrence ross' nationality

LLM Response

```
[PYTHON]
result = df.loc[df['Player'] == 'terrence ross', 'Nationality'].values[0]
[/PYTHON]
```

Ground Truth "united states"

Pandas Query Execution Result "united states" ✓

Figure 1: An example where a LLM (ChatGPT) can generate a correct Pandas query to answer user question using only the table header and column data types, without accessing the table values. Typically, the total number of tokens for DataFrame QA tasks, including both the prompt and the model output, stays below 250 tokens.

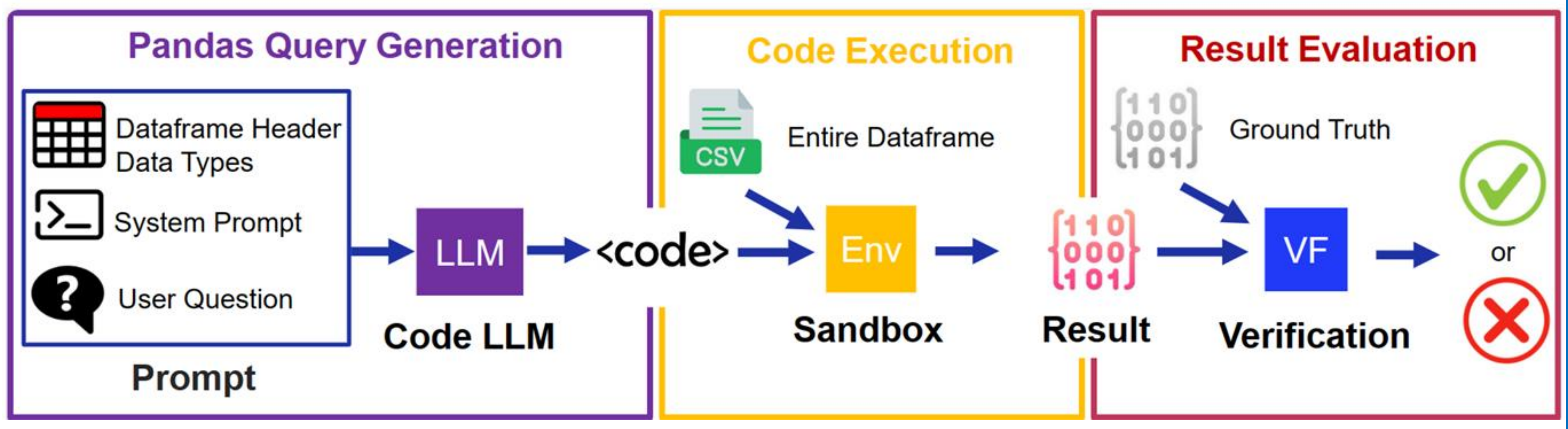
Methodology

Problem Statement:

- Generate executable Pandas quires for answering user's questions on dataframes.
- Use column headers and metadata, ensuring **data security**.

Framework:

1. **Pandas Query Generation**: LLM generates a Pandas queries using dataframe headers, meta data, system prompt, and user questions.
2. **Code Execution**: Pandas queries run in a controlled environment to ensure safety.
3. **Result Evaluation**: Compare results with ground truth for validation.



Pandas Query Generation

Code Execution

Result Evaluation

Figure 2: Framework of DataFrame QA. Note that, LLM in the figure can be replaced with any fine-tuned NLP model trained for the DataFrame QA task.

Datasets

1. WikiSQL (Simple Query Dataset) [1]

- Adapted for DataFrame QA, transforming tables into dataframes.
- Question types: 71% simple retrieval, 29% aggregation tasks.

2. UCI-DataFrameQA (Complex Dataset)

- Develop a DataFrame QA dataset reflecting **real-world scenarios** using GPT4.
- Data sourced from the **UCI dataset** [2], spanning various domains.
- Represent three real-life data interaction roles:
  1. **Data Scientist:** Focus on detailed data analysis (patterns, trends, statistics).
  2. **General Users:** Seek practical, consumer-oriented information.
  3. **Data Owners:** Extract business-oriented insights.

User Question	Pandas Query	Types
which province is bay of islands in?	`result = df.loc[df['Electorate']=='bay of islands', 'Province'].iloc[0]`	Retrieval
how many combined days did go shiozaki have?	`result = df.loc[df['Wrestler']=='go shiozaki', 'Combined days'].values[0]`	Aggregation
how does the average shell weight vary across different numbers of rings?	`result = df.groupby('Rings')['Shell_weight'].mean()`	Data Analysis

can you create a new column 'volume' as a product of length, diameter, and height, then find the average volume for each sex?	`df['Volume'] = df['Length'] * df['Diameter'] * df['Height']`
	`result = df.groupby('Sex')['Volume'].mean()`
		Data Analysis

Table 1: Examples of Sample Questions and Corresponding Pandas Queries Categorized by Complexity Level. Retrieval/Aggregation queries can be resolved using single-step, SQL like queries, whereas Data Analysis questions necessitate multi-step or complex Pandas operations.

Role	Retrieval/Aggregation	Data Analysis
Data Scientist	9 (5%)	175 (95%)
General User	69 (40%)	105 (60%)
Data Owner	42 (22%)	147 (78%)

Table 2: Distribution of Generated Question Types on UCI-DataFrameQA Dataset Across Different Roles.

Experiment and Results

- **GPT-4 Dominance:** GPT-4 demonstrated exceptional performance on both datasets, achieving high accuracy.
- **Scaling Laws [3]:** Larger models generally perform better, but CodeLlama models are an exception.
- **Comparison to Text-to-SQL:** GPT-4's zero-shot approach and the complexity of Pandas queries slightly hinder its performance compared to specialized Text-to-SQL models.

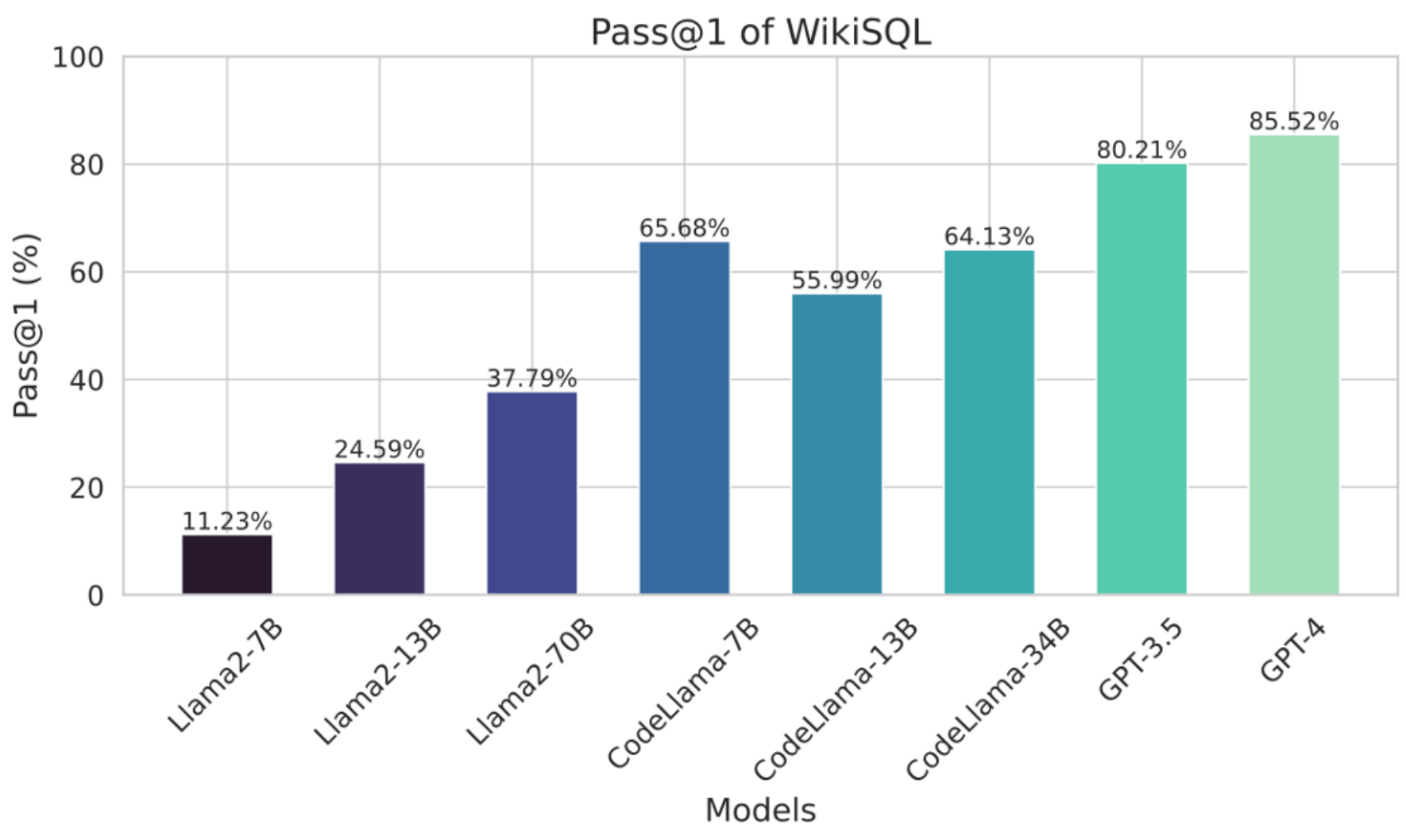


Figure 3: Performance of LLMs on WikiSQL.

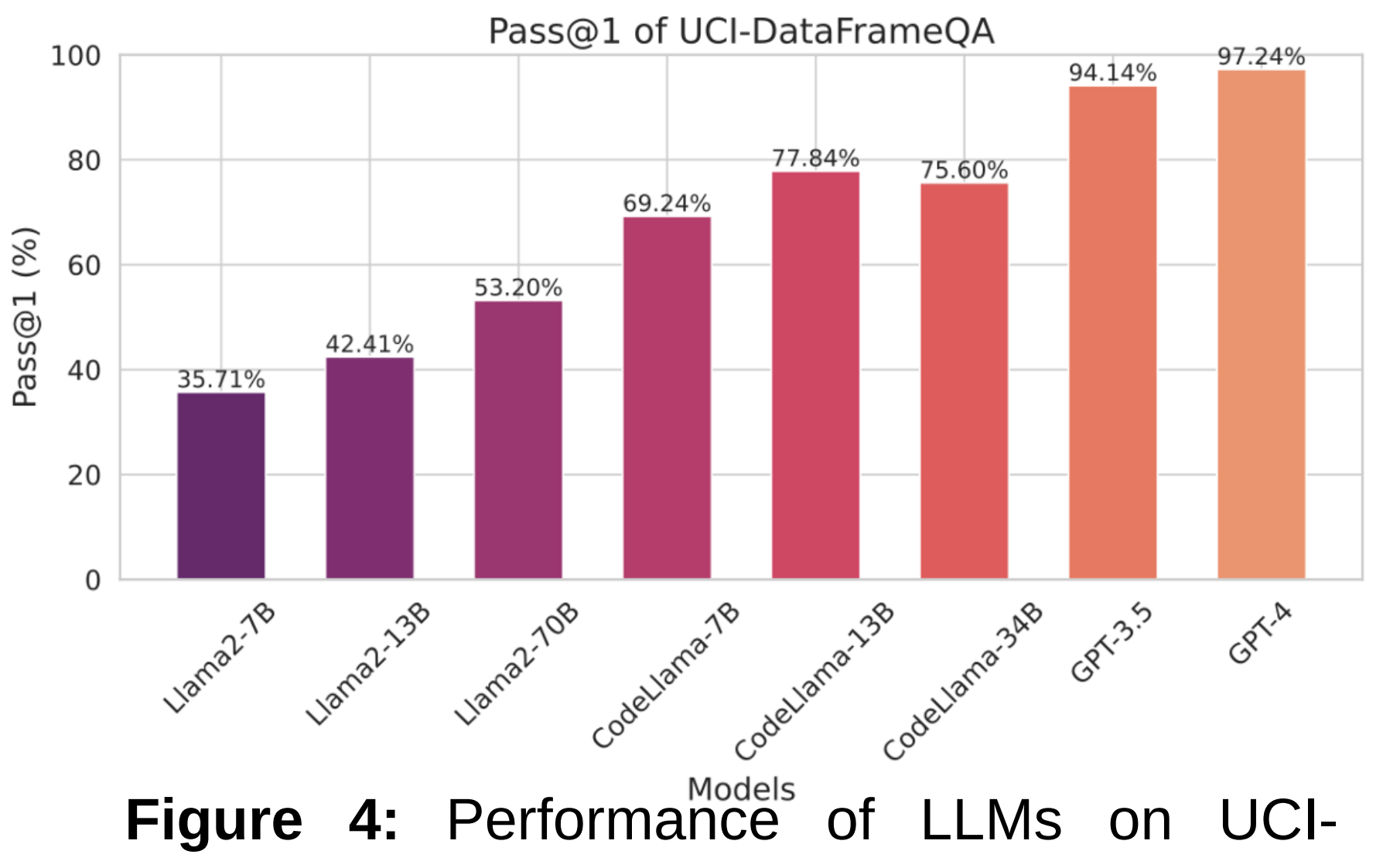


Figure 4: Performance of LLMs on UCI-DataFrameQA.

Error Analysis

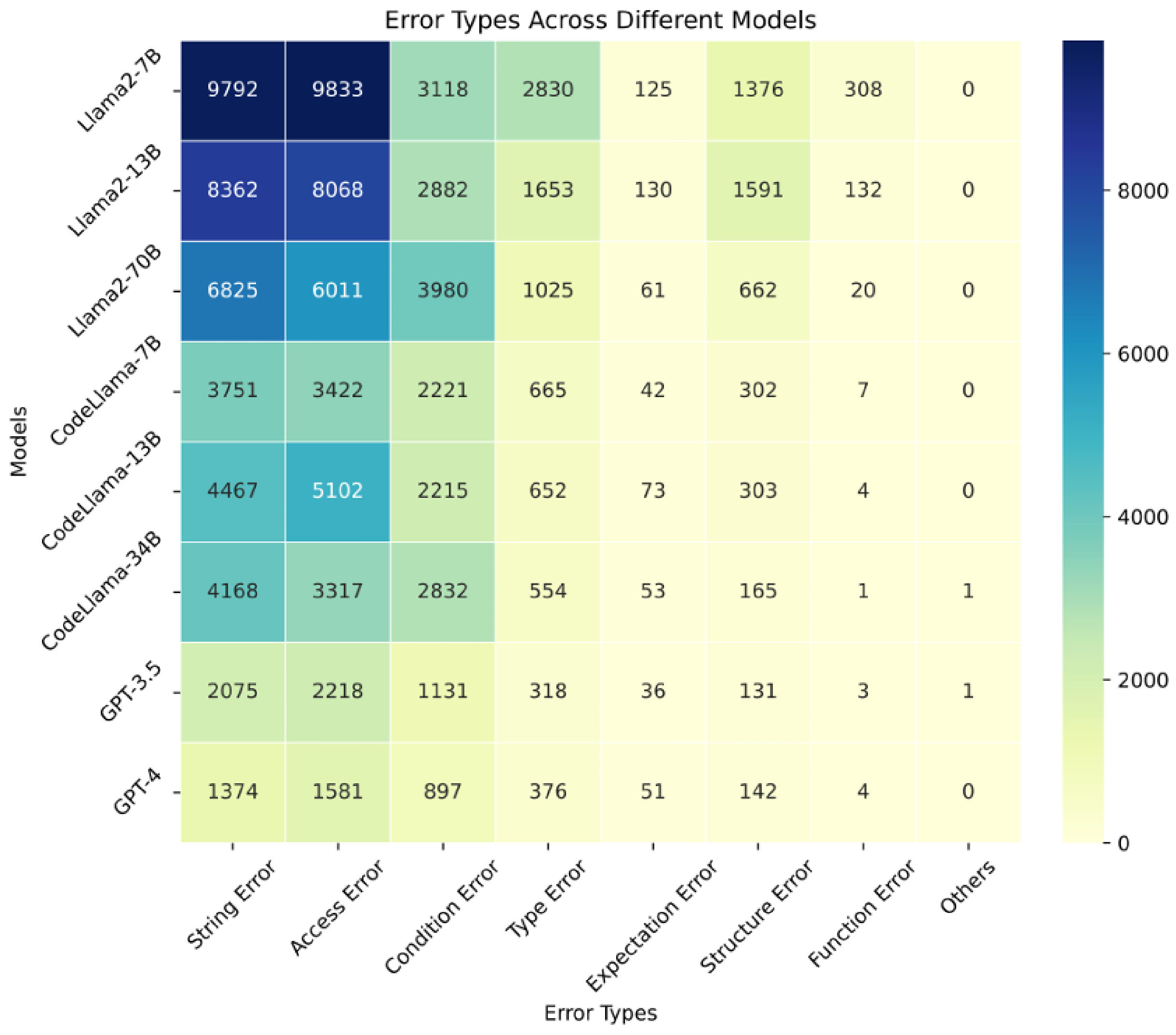


Figure 5: Distribution of Error Types Among Different LLMs on WikiSQL.

Error Type	Eample
Value Retrieval Error	Q : Which province is grey and bell electorate in? P' : `result = df[(df['Electorate']=='grey')|(df['Electorate']=='bell')]['Province'].unique()` P : `result=df[(df['Electorate']=='grey and bell')]['Province'].unique()`
Column Reference Error	Q : What is the variance in resting blood pressure (trestbps) among different heart disease severity levels? P' : `df.groupby('ca')['trestbps'].var()` P : `df.groupby('num')['trestbps'].var()`

Instruction Misalignment	Q : What are the mean and median lengths of abalone by each sex category?
	P' : `import pandas as pd`
	`# Group the dataframe by sex`
	`...`
	P : `result = df.groupby('Sex')['Length'].agg(['mean', 'median'])`

Aggregation Error	Q : what is the total amount of allied-unrelated where the component is human capital? P' `result = df[df['Component']=='human capital']['Allied-Unrelated'].sum()` P `result = df[df['Component']=='human capital']['Allied-Unrelated'].count()`
Function-Column Ambiguity Error	Q : what is the average score when the swimsuit score is 8.503? P' : `result = df[df['Swimsuit']=='8.503']['Average'].mean()` P : `result = df[df['Swimsuit']=='8.503']['Average']`
Insufficient Column Data/Format Information	Q : which team played on december 5? P' `df['Date'] = pd.to_datetime(df['Date'])` `result = df[df['Date'] == 'december 5']['Team']` P : `result = df[df['Date'] == 'december 5']['Team']`
Coding Syntax Error	Q : how does the average shell weight vary across different numbers of rings? P' : `result = df.groupby('Rings').mean()['Shell_weight']` P : `result = df.groupby('Rings')['Shell_weight'].mean()`
Hallucination Error	Q : i'm interested in knowing the most common age of abalone. can you find that for me? P' : `result = df['Age'].mode()` P : `result = df['Rings'].mode()`

Table 3: Typical Failure Cases in DataFrame QA Task. *Q*: User Question, *P'*: Generated Pandas Query, *P*: Correct Pandas Query.

Conclusion

- We introduce **DataFrame QA**, **a secure, zero-shot LLM** framework that leverages dataframe headers to address **data privacy** and **minimize extraneous prompts**.
- By enriching prompts with dataset descriptions, the framework improves performance.
- The success of this task relies on both **coding abilities** and **query comprehension**, with GPT-4 achieving high practical accuracy.

Reference

1. Zhong, Victor, Caiming Xiong, and Richard Socher. "Seq2sql: Generating structured queries from natural language using reinforcement learning." *arXiv preprint arXiv:1709.00103* (2017).
2. Blake, Catherine L. "UCI repository of machine learning databases." http://www. ics. uci. edu/~mlearn/MLRepository. html (1998).
3. Kaplan, Jared, et al. "Scaling laws for neural language models." arXiv preprint arXiv:2001.08361 (2020).