LLM Agents for data engineering code generation

Problem Statement :

LLM agents should be able to generate the code to read the data from the given location and also be able to perform basic tasks of data wrangling such as:

- Group By
- Filter
- Aggregation
- Sorting

This code should be generated in a manner compatible with the specified development environment, such as Databricks, AWS Glue, Google BigQuery.

Project Overview :

The solution uses AutoGen AI agents to generate data engineering code for reading data from files (CSV, Parquet, TSV, XLSX, etc.) and performing data wrangling tasks such as:

- Grouping
- Filtering
- Aggregation
- Sorting

This project leverages two agents:

1. **Programmer Agent**: Generates the required code based on prompts.
2. **Tester Agent**: Analyzes the generated code, detects errors, and provides feedback

Environment Setup :

- Ensure you have Python 3.x installed on your system.
- Install the required packages using requirements.txt.
- Make sure to add your OpenAI API key as an environment variable or directly into your configuration file:

How It Works :

This project defines two AI agents using AutoGen:

- **Programmer Agent:** This agent generates code based on the provided environment and task (e.g., reading data, performing group-by operations).
- **Tester Agent:** This agent evaluates the generated code for syntax and compatibility with the environment and provides suggestions for improvement if necessary.

Code Flow :

- The programmer agent is given a task via a prompt (e.g., "Generate PySpark code that reads a TSV file, filters rows, groups by 'department', and calculates max salary").
- The tester agent evaluates the generated code for errors or issues.If the code passes validation, it is returned; otherwise, the tester provides suggestions for corrections.

Testing with Prompts :

The system has been tested with various prompts, including:

- **Prompt 1**: "Generate PySpark code in Databricks to read a Parquet file, group by 'Region', and return total sales."
- **Prompt 2**: "Generate a SQL query for Snowflake to read from sales_data, group by 'product_id', and return total count."
- **Prompt 3**: "Generate Pandas code to read a CSV file, filter rows where age > 30, group by city, and calculate average income."

For full prompt examples, see the prompts.txt file in the repository.

Contact Us

iSynergy India Office

6, Fourth floor, Rama Pride,
S. No. 118, Near Sarita Nagari,
Behind Axis Bank, Off Sinhagad Road,
Parvati, Pune – 411030,
Maharashtra, India
+91-20-24250337
more.info@isynergytech.com

iSynergy USA Office

2005 Bent Creek Manor,
Alpharetta, GA-30005, USA
+1 (770) 569 7472
more.info@isynergytech.com