Conditional generation of insights on tabular data with LLMs

Wilmer Gonzalez
Universidad Central de Venezuela
`wilmeragsgm@gmail.com`

Abstract

Automated insight generation is a challenging task where a system is asked to provide an interesting observation about a given dataset, this pushes the limitations of subjectivity but would hold a very efficient way of enabling domain experts without data science skills to retrieve observations that can lead to faster decision-making. Previous work has used LLMs partially in stages of such task (Ma et al., 2023) and others have explored a more extensive LLM role in such pursuit (Laradji et al., 2023) but the task remains hard to measure and compare. We explore a methodology for using an LLM in all the stages of insight generation by performing prompt engineering to improve and apply this methodology to three datasets in different domains to try and see its competency. The code and data used is accessible in github[1].

1 Introduction

The word *insight* can be defined as *"an understanding of the true nature of something"* [2], however, we will use the term insights to refer, in particular, to the ones that are *"relevant in the context of a given goal"*, this adaptation is needed to establish a criterion of how an insight might look.

Automated generation of insights based on multi-dimensional data has been explored in different settings involving mechanisms to detect data patterns individually (Ding et al., 2019), retrieving meta-patterns (Ma et al., 2021), that is detecting groups of several patterns, and using Large Language Modelling systems (LLMs from now on) as an orchestration piece (Ma et al., 2023). However, the usage of LLMs has been limited to aspects of the generation process such as being an interface to the user and forming the end statements. We explore the usage of LLMs in most of the stages of insight generation with careful prompt engineering.

To this date, the automated generation of insights has been evaluated manually by human annotators, this suggests the difficulty in measuring performance automatically, as well as the subjective nature of the quality of insights. Similar systems focused on generating consistent statements based on a given dataset, like (Herzig et al., 2020) measure performance with automated metrics, however, this does not cover the capability of generating relevant insights, only its consistency.

On this basis, the central hypotheses of this paper are:

- We expect LLM to be competent to generate insights based on multi-dimensional datasets.
- We expect that data format and prompt engineering practices can improve the quality of insights generation.
- We expect it is possible to create an automated evaluation metric to measure the effectiveness of insights generation systems.

Using LLMs in an end-to-end manner to produce insights based on given multi-dimensional would simplify Exploratory Data Analysis (EDA) activities when performed by domain experts and provide an initial overview that can reduce the time to make decisions in the domain. But to use this type of system, we need a way of measuring performance and manual annotation can be a time-consuming endeavor, a automatically generated metric would assess the quality of the proposed system while being time-efficient.

We experiment with generating automatic insights in two steps described in the Methods section across three different datasets and measure the hit rate of the methodology in producing previously collected insights from EDA practitioners.

In this process, we find that LLMs achieve a low hit rate when compared to the collected insights but qualitative ones do provide interesting insights,

[1] wilmeragsgh/cgi-llms

[2] https://www.britannica.com/dictionary/insight

which indicates a need for defining a more objective or scalable way of defining what is interesting in the context of a given multi-dimensional dataset.

2 Related work

InsightsPilot

(Ma et al., 2023) Propose an automated system that uses an LLM with complementary systems to detect useful information before handing it to the LLM for the final response. Notably, this approach assumes receiving an initial user intent or request, however, proactive general insights can be useful for alerting even when it might be outside of the user's first interest. Additionally, the phrasing of the inquiry might as well affect the capability of such a system to generate insights. This paper also condenses three other major papers that refer to the problem, namely *Quickinsights*, *Metainsights*, and *XInsights* while also extending an efficient representation for insights as data. In our setting, we will explore the limits of capabilities for LLMs in trying to solve the task.

Capture the flag

(Laradji et al., 2023) Proposes an evaluation methodology for agents in the task of recognizing meaningful information, insights to our nomenclature, in a dataset. They also propose two LLMs workflows to test the methodology, in this case, the abstractions are quite similar to the problem setup this literature review expects to deepen. However, this methodology was tested on a single table schema, where we will try to see at least three scenarios.

Fun facts from Wikipedia tables

(Korn et al., 2019) Proposes a framework for deriving interesting facts about Wikipedia superlatives tables, but to date, research methods such as BERT or transformers were not tested, in this work we try a similar task at a much smaller scale for domain-specific tables.

3 Data

The task of generating the insights based on data has typically been evaluated with manual human annotations, as part of our contributions we created a dataset of known datasets along with insights in the same setup described in (Laradji et al., 2023), that is, for each dataset we will have a list of insight considered to be useful. We attempted to use datasets similar to the ones explored in (Ding et al., 2019), however, we prioritized having EDA sessions that can serve as reference insights for the used datasets.

The following datasets were collected from the platform Kaggle to pair them with insights collected from EDA sessions worked by the community:

student-study-performance

This dataset consists of the marks secured by the students in various subjects and was paired with 4 insights. See columns description appendix A. An exemplary insight for this dataset is *"The highest average is confined to students whose parents are master degree holders"*.

global-air-pollution-dataset

This dataset provides geolocated information about pollutants and was paired with 4 insights. See columns description appendix B. An exemplary insight for this dataset is: *"India notably has a substantial number of cities with high AQI values, indicating severe pollution issues, while Brazil stands out for having the most cities with good air quality"*.

auto-sales-data

This dataset contains sales data of an Automobile company and was paired with 14 insights. See columns description appendix C. An exemplary insight for this dataset is: *"United States leads in sales, followed by Spain and France. Notable sales from Australia and Singapore in the Eastern Hemisphere"*.

4 Model

The modeling approach we will use resembles the one proposed in (Laradji et al., 2023) specifically the explorer agent (as described in E), however, we disclose the implementation used, that is using LangChain[3] instead of DSPy to use the Pandas agent that can help speed up the process on interacting with the data through python code. In concrete, we use OpenAI's *'gpt-3.5-turbo-instruct'* with temperature=0 to ensure consistency of results. In our implementation, we used a custom version of the prompt in which we provide exemplary questions

[3] langchain.com

that can lead to insights, following the types of insights described in [4]

5 Methods

Our approach was composed of the following steps being replicated to each dataset:

1. Select a reasonable goal for the insights.
2. Extract the column names of the dataset.
3. Provide previous steps results to the LLM in a chat setting to request questions that can help find insights.
4. Ask the questions generated to LLM by using pandas dataframe to use python code as intermediary representations to answer the questions [5]
5. Measure the hit/miss rate of the insights paired with the dataset initially with the ones obtained.

6 Results

This section will be organized per each dataset tested

auto-sales-data

From the insights paired with the dataset, this methodology was able to capture:

- The USA has the highest total sales with 3355575.69
- Classic Cars has the highest quantity ordered with 33373 units.
- The most common deal size in this dataset is Medium.

This results in **0.214** hit rate on the expected insights based.

It is noticeable that some questions generated in step 4. would not lead to an informative insight, for example, to the question *'Is there a trend in the sales amount over time?'* the answer was: *'Yes, there is a trend in the sales amount over time.'*

global-air-pollution-dataset

From the insights paired with the dataset, this methodology was able to capture:

- There is a positive correlation between PM2.5 AQI Value and NO2 AQI Value.
- The countries with consistently high AQI Values throughout the year are Republic of Korea, Bahrain, Mauritania, Pakistan, United Arab Emirates, Aruba, Kuwait, Qatar, India, Senegal, Saudi Arabia, Gambia, Yemen, Guinea-Bissau, Oman, China, Kingdom of Eswatini, Uzbekistan, Nepal, Tajikistan, Democratic Republic of the Congo, Iraq, Bangladesh, South Africa, Iran (Islamic Republic of), Dominican Republic, Libya, Turkmenistan, Haiti, and Egypt.[6]

This results in **0.25** hit rate on the expected insights based.

student-study-performance

From the insights paired with the dataset, this methodology was not able to capture any insight as is, however upon further inspection one of the paired insights was better interpreted by the LLM, in concrete we had the insight *The highest average is confined to students whose parents are master degree holders*, but the data shows just a slight edge over parents with other degrees. Because of this, it seems valid to make the statement that there is no pattern there, which the model captures by generating the question *'How does parental level of education affect test scores?'*, but answering *'The parental level of education does not seem to have a significant effect on test scores, as the mean scores for each group are relatively similar.'*

7 Analysis

The overall performance in terms of hit rate of **0.15** Indicates there is room for improvement in both the modeling and data collection practices, it also suggests that it might be feasible to use this methodology to produce initial observations about the data. The scale of pairs of (datasets, and insights) could help improve the quality of systems in this task. The hit rate was used as a measure after understanding the inherently subjective nature of how interesting an insight is, but this area can be improved as well.

[4] microsoft.com/../Insight-Types-Specification

[5] langchain.com/../pandas/

[6] The insight initially only mentioned India which make this insight not completely a hit

8 Conclusion

In the face of realizing the complex nature of insights, **we find that is hard to measure the quality of a system trying to generate automated insights by the methodology presented** and, in consequence **it is highly subjective if LLMs can be competent in generating insights for multi-dimensional datasets**. However, in further review of similar approaches such as (Korn et al., 2019) we might find a way of scaling the capability of newer systems that were not tested during such research. We also found integrations in LLM toolkits such as LangChain/Pandas highly useful when working in this setup, and these aspects might hold future success because of their efficiency in handling data. Following the example of (Wenhu Chen and Wang, 2020) it might be useful to collect a large-scale dataset of tables with subspaces of data or questions that might lead to interesting findings.

Known Project Limitations

Initially, the research ideas aimed to create a standard workflow for using LLM systems to automatically generate insights, and provide an evaluation criterion on such task. Even during the literature review and experimental protocol designing, it seemed feasible to build on top of existing research, that omitted this standardization, by addressing the subjective nature of *'what is an insight'* with the creation of reliable evaluation metrics. However, after further review and reflection on the needs of a hypothetical domain expert looking for insights retrieved from a given dataset, it became clear that such a setting to generate insights can only be subjective, to allow the domain expert to establish a goal after which a given insight can be considered interesting or useful. Because of this subjectivity, the workflow tested during this article is prone to have a different performance given a different domain and goal of a domain expert.

The context of the LLM would limit the number of columns that can be used in a single pass, this column information is needed to establish the schema and provide it to the LLM so it generates questions of the insights to look for. Because of this, the LLM setup can only be applied to small datasets with the cardinality of columns being limited to the context size.

Authorship Statement

This research article is the sole work of the author.

References

Rui Ding, Shi Han, Yong Xu, Haidong Zhang, and Dongmei Zhang. 2019. Quickinsights: Quick and automatic discovery of insights from multi-dimensional data. *Proceedings of the 2019 International Conference on Management of Data*.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. TaPas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333, Online. Association for Computational Linguistics.

Flip Korn, Xuezhi Wang, You Wu, and Cong Yu. 2019. Automatically generating interesting facts from wikipedia tables. In *Proceedings of the 2019 International Conference on Management of Data*, SIGMOD '19, page 349–361, New York, NY, USA. Association for Computing Machinery.

Issam Hadj Laradji, Perouz Taslakian, Sai Rajeswar, Valentina Zantedeschi, Alexandre Lacoste, Nicolas Chapados, David Vazquez, Christopher Pal, and Alexandre Drouin. 2023. Capture the flag: Uncovering data insights with large language models. *ArXiv*, abs/2312.13876.

Ping Ma, Rui Ding, Shi Han, and Dongmei Zhang. 2021. Metainsight: Automatic discovery of structured knowledge for exploratory data analysis. *Proceedings of the 2021 International Conference on Management of Data*.

Pingchuan Ma, Rui Ding, Shuai Wang, Shi Han, and Dongmei Zhang. 2023. InsightPilot: An LLM-empowered automated data exploration system. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 346–352, Singapore. Association for Computational Linguistics.

Jianshu Chen Yunkai Zhang Hong Wang Shiyang Li Xiyou Zhou Wenhu Chen, Hongmin Wang and William Yang Wang. 2020. Tabfact : A large-scale dataset for table-based fact verification. In *International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia.

A Columns for student-study-performance

Column Name: Description

gender: sex of students

race/ethnicity: ethnicity of students.

parental level of education: parents' final education.

lunch: having lunch before test.

test preparation course: complete or not complete before test.

math score: score points for math test.

reading score: score points for reading test.

writing score: score points for writing test.

B Columns for global-air-pollution

Column Name: Description

Country: Name of the country.

City: Name of the city.

AQI Value: Overall AQI value of the city.

AQI Category: Overall AQI category of the city.

CO AQI Value: AQI value of Carbon Monoxide of the city.

CO AQI Category: AQI category of Carbon Monoxide of the city.

Ozone AQI Value: AQI value of Ozone of the city.

Ozone AQI Category: AQI category of Ozone of the city.

NO2 AQI Value: AQI value of Nitrogen Dioxide of the city.

NO2 AQI Category: AQI category of Nitrogen Dioxide of the city.

PM2.5 AQI Value: AQI value of Particulate Matter with a diameter of 2.5 micrometers or less of the city.

PM2.5 AQI Category: AQI category of Particulate Matter with a diameter of 2.5 micrometers or less of the city.

C Columns for auto-sales

Column Name: Description

QUANTITYORDERED: It indicates the number of items ordered in each order.

PRICEEACH: This column specifies the price of each item in the order.

ORDERLINENUMBER: It represents the line number of each item within an order.

SALES: This column denotes the total sales amount for each order, which is calculated by multiplying the quantity ordered by the price of each item.

ORDERDATE: It denotes the date on which the order was placed.

DAYS_SINCE_LASTORDER: This column represents the number of days that have passed since the last order for each customer. It can be used to analyze customer purchasing patterns.

STATUS: It indicates the status of the order, such as "Shipped," "In Process," "Cancelled," "Disputed," "On Hold," or "Resolved."

PRODUCTLINE: This column specifies the product line categories to which each item belongs.

MSRP: It stands for Manufacturer's Suggested Retail Price and represents the suggested selling price for each item. PRODUCTCODE This column represents the unique code assigned to each product.

CUSTOMERNAME: It denotes the name of the customer who placed the order.

PHONE: This column contains the contact phone number for the customer.

ADDRESSLINE1: It represents the first line of the customer's address.

CITY: This column specifies the city where the customer is located.

POSTALCODE: It denotes the postal code or ZIP code associated with the customer's address.

COUNTRY: This column indicates the country where the customer is located.

CONTACTLASTNAME: It represents the last name of the contact person associated with the customer.

CONTACTFIRSTNAME: This column denotes the first name of the contact person associated with the customer.

DEALSIZE: It indicates the size of the deal or order, which are the categories "Small," "Medium," or "Large."

D Prompt used for insights questions generation

"Generate 10 questions in a JSON array about the following dataset using all the columns that help find answers to insightful questions about {goal}. Columns: {columns} Examples of insightful questions: 'What is the dimension value with the highest value in this [data_view]?', 'What is the dimension value with the lowest value in this [data_view]?', 'Is there an increasing pattern in this [data_view] in time for the group X?', 'Is there an decreasing pattern in this [data_view] in time for the group X?', 'What is the outlier value in this [data_view]?', 'What is the period of seasonality in this [data_view]?', 'What is the valley in this [data_view] with respect to time?', 'What is the peak in this [data_view] with respect to time?'.

{format_instructions}"

E Algorithm for explorer agent

```
Require: rawData, generalGoal, nRounds, dataContext
Ensure: Top insights about the dataset
Initialize: questions <- {}, answers <- {}, insights <- {}
while not Reached(nRounds) do
    Stage 1: Generate questions using the goal and the accumulated insights
    questions <- AskLLM(rawData, generalGoal, insights, dataContext)
    //as given by the prompt at App. A.1.1
    Stage 2: Generate code for each question
    for each question in questions do
        code <- GenerateCodeToAnswer(question, dataContext)
        answer <- ExecuteCode(data, code)
        answers.add(answer)
    end for
    Stage 3: Extract insights from answers
    insights.add(ExtractInsights(answers))
end while
Return TopInsights(insights)
// as given by the prompt at App. A.1.2
```