# Pipe*(line)* Dreams: Fully Automated End-to-End Analysis and Visualization

Cole Beasley
cole.beasley@nyu.edu
New York University Abu Dhabi
Abu Dhabi, UAE

Azza Abouzied
azza@nyu.edu
New York University Abu Dhabi
Abu Dhabi, UAE

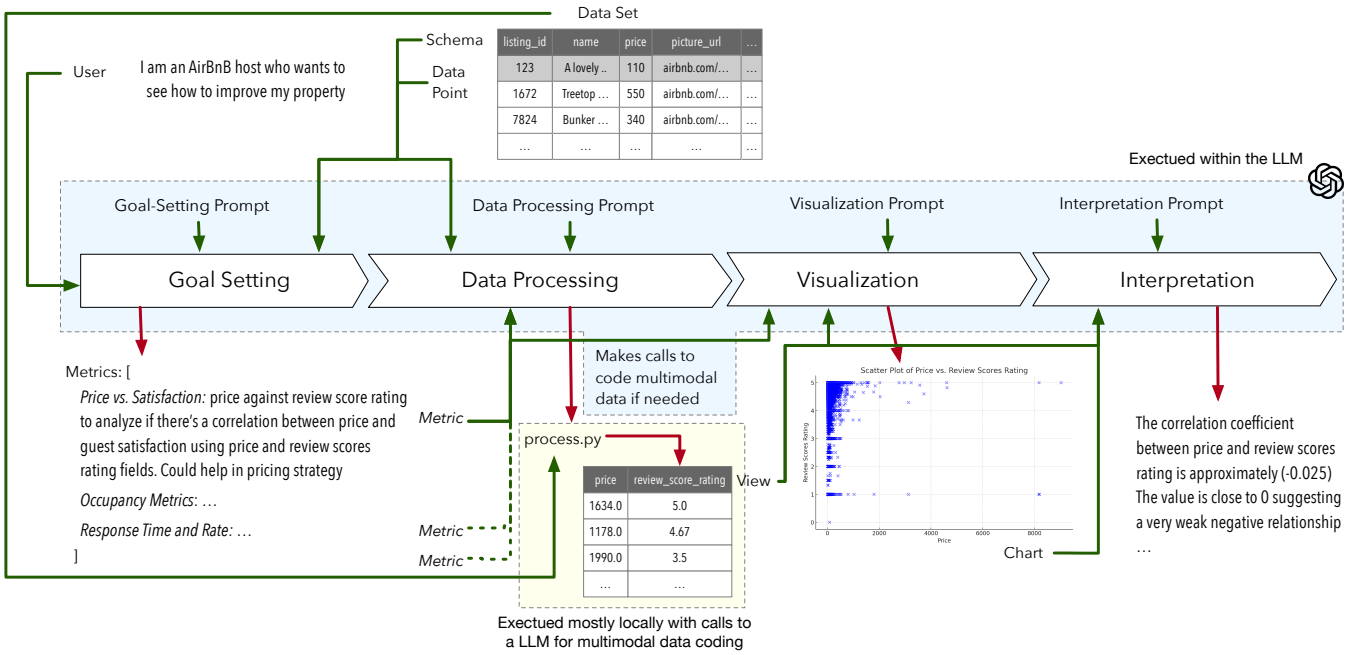



**Figure 1: LLM4Vis's four steps of automated end-to-end analysis implemented as LLM nodes. Green arrows → indicate inputs to each LLM node and red arrows → indicate outputs.**

## ABSTRACT

We exploit large language models (LLMs) to automate the end-to-end process of descriptive analytics and visualization. A user simply declares who they are and provides their data set. Our tool LLM4Vis *sets* analysis *goals* or metrics, generates code to *process* and analyze the data, *visualizes* the results and *interprets* the visualization to summarize key takeaways for our user. We examine the power of LLMs in democratizing data science for the non-technical user and in handling rich, multimodal data sets. We also explore LLM4Vis's limitations, opportunities for human-in-the-loop interventions, and challenges to measuring and improving the robustness and the utility of LLM-generated end-to-end data analysis pipelines.


*HILDA 24 , June 14, 2024, Santiago, AA, Chile*

ACM ISBN 979-8-4007-0693-6/24/06
https://doi.org/10.1145/3665939.3665962

**ACM Reference Format:**
Cole Beasley and Azza Abouzied. 2024. Pipe*(line)* Dreams: Fully Automated End-to-End Analysis and Visualization. In *Workshop on Human-In-the-Loop Data Analytics (HILDA 24 ), June 14, 2024, Santiago, AA, Chile.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3665939.3665962

## 1 INTRODUCTION

For whatever one does, there is probably data out there of some value. An enterprising stakeholder asks "*What can I learn from this data?*" After all, there is ample evidence of data put to good, and sometimes unexpected, use across a wide-range of domains. Yet, transforming raw data into an artifact such as a visualization that provides insight, is complex and time-consuming, and requires a technical skill set that is often distinct from a typical stakeholder's domain expertise. And so, democratizing — narrowly defined here as cutting-down the required statistical, mathematical, programming and visualization know-how — this effort (and data science in general) is an active and ongoing research area [17]. So far, automated systems with or without human-in-the-loop interventions have focused on specific parts of this end-to-end analysis pipeline but not on automating its entirety: from model selection [18] to

the automated visualization of specific data fields [14]; Systems that do explore end-to-end automation require clear specifications of the task and do not support automated and autonomous task specification or goal-setting (see for e.g. [17]).

In this workshop paper, we explore the possibility of a system where a stakeholder provides a dataset and simply states "*I am a ... I have this data ... What can I learn from it?*" The system then conducts the end-to-end goal-setting, processing, analysis, visualization and interpretation to provide an answer. Our focus is on providing descriptive answers rather than predictive ones (e.g. autoML systems), or prescriptive ones.

We argue that the emergence and growing maturity of large language models (LLMs) makes this vision achievable, and using a case study with AirBnB data and four different stakeholders, we show promising evidence of this.

*But why* do we think that LLMs can make this vision more attainable? Researchers have long recognized the relationship between effective visualizations and storytelling [11, 15, 16]. Journalists have used data narratives or stories to powerfully convey their message and influence readers[1]. This relationship between stories and effective visualization makes LLMs particularly apt for end-to-end analysis. Language models generate the most probable sequence of tokens (e.g. words, sentences) given initial prompts; Prompted by what a person does, they can generate highly-probable motivators and drivers for them in a narrative sense; from these drivers, they can generate highly-probable metrics, which in turn determines what to look for in a dataset, and how to visually convey these metrics. Moreover, with the ability to closely match human preferences in qualitative data interpretation, such as the rating or classification of images, these generated end-to-end pipelines can not only integrate multimodal data sets but can also define what can be extracted from them that is meaningful [23, 24].

*And how* do we build such as system? We present in this paper a *sequential prompting* approach for end-to-end descriptive analysis (§2). The initial and only user-provided prompt simply describes the stakeholder and the data. From then on, our system, LLM4Vis, takes over. First, in a *goal setting* step, it identifies several relevant and viable metrics to extract and analyze from the data. The outputs of this step are sequentially fed as prompts into a *data processing* step, which generates a script to extract and derive appropriate attributes. The reduced data set from executing the script is then sequentially fed into a *visualization and interpretation* step that produces the final visualization output and a note on what can be learned from it. LLM4Vis integrates the outputs of one step into template prompts to form inputs for the next step (Figure 1). This approach opens up the space for human-in-the-loop interventions such as *expansion* (e.g. adding more data), and *refinement* (e.g. scoping down or redefining the goals, fields of interest or metrics). We discuss this along with other research questions and challenges (§5), but we begin with the most straightforward analysis: *can this work?* exposing along the way limitations and areas of improvement (§3).

***Exemplar Case Study.*** Consider the publicly available data sets of AirBnB *short-term vacation rentals* downloadable from providers like Inside AirBnB [3]. This is a rich data set[2]. It has spatial and temporal fields (location, rent prices or availability of over time). It has multimodal components: images of the units, unstructured text descriptions, etc), along with numerical and categorical data values (occupancy, size, etc.)

We simulate four stakeholders[3]: a current *host* who is trying to maximize profits from their current listing; an *investor* who wants to understand what features of a property make for a good rental; a *city tourism board* that wants to better understand the relationship between short-term rentals and tourism in the city; and a *photographer* that wants to understand how to best capture a client's property.

## 2 AN OVERVIEW OF LLM4VIS 1.0

Figure 1 illustrates how after a user provides a sample of their data set (schema and one data point), a sequence of LLM nodes sequentially process templated prompts instantiated by inputs from the previous node to feed their outputs to the next node. Each LLM node codifies a task within the automated end-to-end analysis pipeline. Exact output format specifications simplify downstream prompts and processes that ingest this output. Breaking down the end-to-end analysis process into a series of tasks is a more effective and easier-to-debug method for guiding LLMs through complex reasoning tasks [22].

### 2.1 Sequential Prompting

*2.1.1 Step 1: Goal Setting.* Using the prompt below, the goal-setting node in Figure 1 leverages a LLM to determine *metrics* that are not only *relevant* to the user's end-goals but are also *viable* (i.e. can be extracted from the data set). Table 2 lists some of the metrics generated for the four stakeholders.

> ***Goal Setting Prompt.*** You are generating metrics that will be displayed in a dashboard. Users specify as input: a *description* of who they are, a *data schema*, and a *data point* from the whole data set. You should be able to derive the metrics from the data. Output your result as a JSON array called 'metrics'.
>
> ---
>
> - *User*: "I am an AirBnB host who wants to see how to improve my property"
> - *Schema*: [listing_id, name, price, picture_url, ... ]
> - *Data Point*: [123, A lovely room in the.. , $110, ...]

*2.1.2 Step 2: Data Processing.* Using the prompt below, for each metric generated in the previous step, the data-processing node generates a Python script that *selects*, *aggregates* and *projects* a view of the data set. The schema and the sample data point ground this step. The script is executed locally except when processing multimodal data; we make LLM API calls to label such data. LLM code generation, rather than directly processing the data by a LLM, has three benefits: (i) it reduces LLM computational costs from repeated data parsing or projection, and aggregation which do not require generative power, (ii) it reduces the amount of tokens passed

[1] https://www.nytimes.com/spotlight/graphics

[2] We focus on one dataset for brevity, but we tested LLM4Vis with other rich urban data sets with similar results.

[3] We interviewed representatives from each of these stakeholder groups to understand their motivations, what they do and how (or if) they would use such a data set.

back and forth by never passing the full data — many LLMs limit the maximum number of tokens or charge per token (iii) it allows users to work with proprietary or private data sets that they many not wish to upload to a third party LLM.

> ***Data Processing Prompt.*** Your job is to make a Python script, which will pick relevant data out of a CSV file. I will give you *a metric* which I want the data for, a *schema*, and a *data point*. The script should be one which I can later run to load all the data. Format your results into a CSV file called 'view.csv'. If any of the needed data points require qualitative coding, such as image or textual analysis, be sure to include the needed calls to the GPT API
>
> ---
>
> - *Metric*: "Price vs. Satisfaction: ... analyze if there's a correlation between price and guest satisfaction using price and review_scores_rating. Could help in pricing strategy."
> - *Schema*: [listing_id, name, price, picture_url, ... ]
> - *Data Point*: [123, A lovely room in the.. , $110, ...]

*2.1.3 Step 3: Visualization.* Using the prompt below, the visualization node creates a graphic from the *metric* specification generated in Step 1 and the *view* from the previous step. GPT4 can execute code with the 'Code Interpreter' feature: the model generates a script using plotting libraries (e.g. PyPlot), executes it and outputs a graphic.

> ***Visualization Prompt.***Your job is to create the best chart to present a *metric* on the given data *view*. Make sure to follow best practices for making charts.
>
> ---
>
> - *Metric*: "Price vs. Satisfaction"
> - *View*: "[price, review_scores_rating
>  1634.0, 5 ...]"

*2.1.4 Step 4: Interpretation.* Using the prompt below, the interpretation node uses code execution to output a brief synopsis of the visualization, which can include analytical values such as statistical significance of trends, and key insights if any.

> ***Interpretation Prompt.*** You will be given a data view and a graphic chart. Your job is to analyze this data and chart, and identify any key insights. Give a broad overview of the information being presented in addition to takeaways found.
>
> ---
>
> - *View*: "[price, review_scores_rating
>  1634.0, 5 ...]"
> - *Chart*: <chart.png>

*Implementation Specifics.* We use OpenAI Assistant [1] for each node. LLM4Vis *chains* the assistants together to allow for sequential prompting. Both visualization and interpretation utilize 'Code Interpreter', GPT's support for executing code within the LLM.

To improve LLM4Vis's performance, we also expand each prompt with *chain-of-thought* and *few-shot* prompting techniques [20]. For example, by expanding the goal-setting prompt as follows, we improved metric relevance and viability by 27%:

> A sample entry for "metrics" if the user was a used car salesman, and the data was used car sales, could be as follows: "Average sale price per brand: The average sale price per brand gives a breakdown of the value differences between cars. This is found by looking at the sale_price and car_brand fields"

Finally, we have not established any ***guardrails*** to protect against misleading analysis, visualizations or interpretations. The importance of guardrails cannot be overstated. Our preliminary evaluation (§3) while promising alludes to the difficulty of implementing fail-safe guardrails especially as easy-to-verify requirements such as viability are often ignored.

## 3 PRELIMINARY EVALUATION

*Setup.* For each of the four stakeholders, we describe the results of one execution run of LLM4Vis with the AirBnB data set (§1). Repeated runs generate slightly different metrics and hence visualizations, given the generative nature of LLMs, but overall our findings stay the same.

*Overview.* Table 1 shows a high-level summary of the four stakeholders' runs. It shows the number of metrics generated that depend on primitive data types (e.g. price) or multimodal data (e.g. review text or property images) and, it assesses the output quality at every step. Eventually, only 21% of the metrics result in insightful visualizations. Generating more metrics initially leads to more useful visualizations and insights as each step fails to proceed for some of its inputs. We describe how in the following sections.

*Goal Setting & Metrics.* Table 2 shows a sample of metrics generated per stakeholder. The metrics are distinct and illustrate the model's ability to distinguish differences across the stakeholders' motivations and generate metrics accordingly.

Across the four stakeholder runs, a total of 79 metrics were generated. Of these

- 91% were *relevant*: i.e. addressed a stakeholder's need and understood data semantics
- 75% were *viable*: i.e. relied on fields that exist or can be derived from the data without external data

A metric like *impact of tourism on local businesses* while relevant to the tourism board is not viable because there is no data about local businesses in the provided data set. A metric like *booking lead time*, which LLM4Vis describes as analyzing the differences between last_scraped and min_nights, max_nights is irrelevant because the fields it determines the metric from have nothing to do with booking lead time: the model either misinterpreted the data semantics or the metric itself, hence generating nonsense.

*Data Processing.* Only 66% of the viable metrics were processed correctly. Failures were mainly due to data parsing or extraction errors (29.8%) — e.g. misgrouping categories due to case sensitivity, mishandling missing values — with only 10% due to incorrectly formatting the output view.

The recent integration of multimodal data interpretation within LLMs may explain the relatively higher failure rates (57%) incorrectly processing metrics that use multimodal data: errors were more basic in nature such as calling the wrong API/model for coding or classifying images. As the technology matures, we expect our

| | | *Goal Setting* | | | *Data Processing* | | *Visualization* | *Interpretation* | |
|---|---|---|---|---|---|---|---|---|---|
| ***Stakeholder*** | Data Type | Metrics Generated | Relevant? | Viable? | Correct Extraction? | Correct Formatting? | Apt Visual Encoding? | Descriptive? | Insightful? |
| Host | Primitive | 27 | 22 | 25 | 18 | 15 | 14 | 14 | 6 |
| | Multimodal | - | - | - | - | - | - | - | - |
| Investor | Primitive | 24 | 24 | 12 | 10 | 10 | 10 | 8 | 5 |
| | Multimodal | - | - | - | - | - | - | - | - |
| Tourism Board | Primitive | 19 | 18 | 14 | 9 | 8 | 8 | 8 | 4 |
| | Multimodal | 2 | 1 | 1 | - | - | - | - | - |
| Photographer | Primitive | 1 | 1 | 1 | - | - | - | - | - |
| | Multimodal | 6 | 6 | 6 | 3 | 3 | 3 | 3 | 2 |

**Table 1: For each stakeholder, LLM4Vis generates several metrics from fields that should exist in the data that are either primitive (e.g. numerical, categorical, ordered, etc.) or multimodal. From these initial metrics, we show how many eventually lead to meaningful visualizations and insights.**

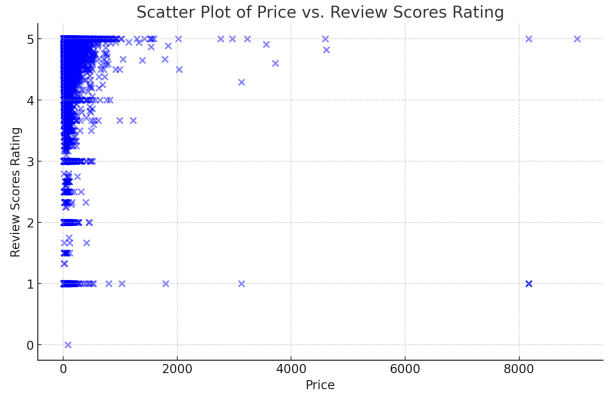
| **Metric** | **Description** |
|---|---|
| ***Host**: "I am an airbnb host who wants to see how to improve my property's performance"* | |
| Price vs. Satisfaction | Scatter plot of price against review_scores_rating to analyze if there's a correlation between price and guest satisfaction. Could help in pricing strategy. |
| Response Time Analysis | Scatter plot showing the distribution of host response rates, based on the host_response_rate field. Quick response times might correlate with higher review scores. |
| ***Investor**: I am thinking of investing in a property and want to see if my initial investment will pay off* | |
| Pricing Strategy | Review price in correlation with the num. of bedrooms, beds, and amenities offered. Compare with similar listings in neighborhood to ensure competitive pricing while maximizing your revenue. |
| Average Daily Rate | Derive from price to understand average earning potential per day. |
| ***Tourism Board**: I work for a city tourism board and want to see the impacts of AirBnB on the city* | |
| Property Type Diversity | Break down listings by property_type and room_type to assess accommodation diversity offered to tourists. |
| Review sentiment analysis | Utilize textual analysis on the description and reviews fields to generate a sentiment score for each listing. Could identify how positive or negative they are overall. |
| ***Photographer**: I am taking photos for an AirBnB client and want to know what things about the photos help properties the most* | |
| Impact of Photo Quality on Bookings | Analyze relationship between quality of property photos and booking frequency. Examine picture quality and professionalism found in picture_url and correlate with reviews_per_month. |
| Photo Themes and Neighborhood Preference | Categorize photos by theme (e.g., modern, rustic, vintage) with image analysis on picture_url to and compare to popularity different neighborhoods (neighbourhood_cleansed) to see if certain themes perform better in specific areas. |

**Table 2: A sample of metrics and their descriptions for the four stakeholders.**

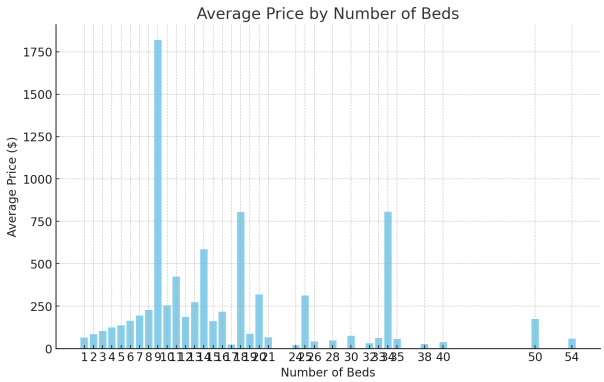
failure rates to drop. When it did work, multimodal data was coded quickly. For example, thousands of property photos were rated reliably enough in terms of quality to yield a sensible visualization, Figure 2(d), of the relationship between photo quality and number of reviews (an approximation of number of bookings, which does not exist in the data).

*Visualization.* We evaluate the final graphic on objective criteria such as the application of appropriate data transformations, the use of apt visual encoding given data type, the correct labeling of axes, etc. without assessing the subjective effectiveness of the visualization. The one failure here was due to an internal error in the model.

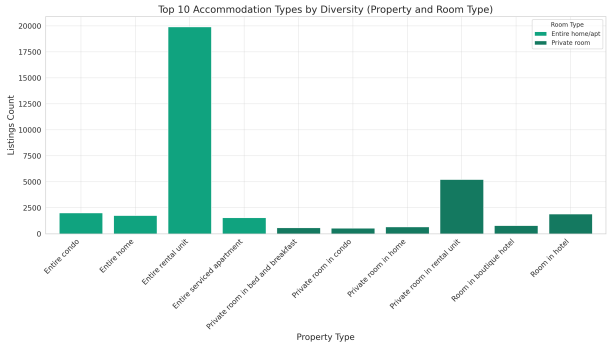
*Interpretation.* We found the analysis produced in the interpretation stage to be very dependent on whether additional statistics were derived. For example, for Figure 2c surface level observations were only provided including: *"The 'Entire rental unit' category is by*




**(a) Host: Price vs. Satisfaction**




**(b) Investor: Pricing Strategy**




**(c) Tourism Board: Property Type Diversity**


**(d) Photographer: Impact of Photo Quality on Bookings**

**Figure 2: Four final visualizations produced by the pipeline for the separate stakeholders. See Table 2 for LLM4Vis's description of each metric.**

*far the most prevalent accommodation type, predominantly offered as 'Entire home/apt'"*. For Figure 2a however, the LLM ran additional scripts yielding a much more developed analytical response: *"Despite the weak correlation, the p-value associated with this correlation ((p = 0.000188)) suggests that the observed correlation is statistically significant. It means that, although the correlation is weak, it is unlikely to have occurred by chance. However, the practical significance of this correlation, given its magnitude, might be limited."*.

## 4 HUMAN-IN-THE LOOP & FEEDBACK LOOPS

To our question *Can this work?*, we say *YES!* With no human intervention we find the 21% success rate of producing meaningful charts and takeaways to be quite promising.

To handle irrelevant or non-viable metrics, users can examine the list of metrics and mark them as problematic, allowing the model to regenerate new or modify these metrics (e.g. choose alternative fields that approximate non-existent ones). This *refinement* however does require users to be familiar with the data schema: a strong assumption for casual, non-technical users. The user's selection of metrics can also guide the model to suggest related areas of interest and hone in on interesting and usable metrics quicker. In cases where the metrics are relevant but require additional data, one can imagine minimal user labeling, which can trigger an automated search-and-data-retrieval process to *expand* the data set. Recent research on Retrieval Augmented Generation (RAG) for LLMs can inform this process [10].

Users could prod for details or more specific analysis: *Why is the average price of a 9-bedroom rental more than 7 times that of an 8-bedroom?* (Figure 2b) As we consider human interventions, we ought to also ask how much of what a human can do can be replaced by an agent [5] that validates rather than generates to provide immediate feedback at each step. Systematic checks around common data transformation errors (e.g. mishandling missing data, incorrect date transformations) may be better handled by automated agents rather than humans. Unsupervised clustering techniques can be used, for example, to assess the quality of multi-modal data codes. AI agents can also act on behalf of a specific stakeholder's profile. For example, acting on behalf of a veteran investor, it can provide feedback to drop and redo the end-to-end analysis until it arrives at a surprising, high-risk, high-reward investment finding. For a beginner, it may provide more essential information such

as the relationship between property size and revenue. Such an agent can also holistically examine the diversity of a collection of generated visualizations to diversify the findings presented to a user in a dashboard.

To enable more actionable feedback from either human- or AI-agents, however, we may need to redesign LLM4Vis's intermediate outputs. In particular, to allow users to point out semantic errors in data processing or transformation, we need to produce scripts in a higher-level data transformation language, which consists of a few well-defined operators or be declarative in nature. A higher-level language can enable the combination of logical program-synthesis techniques with the generative power of LLMs to more consistently create scripts that are easier to verify for correctness. Users should also be able to inspect and correct if necessary the views produced by the data processing step, as well as ground-truth a sample of the multi-modal data coding outputs — such feedback can be used to retrain data coding pipelines. Beyond the graphic produced by the visualization step, generating a visualization script in a higher-level domain-specific language can allow users to critique the effectiveness of certain visualization choices such as opacity, density-reduction, bucketing, scales, use of trend-lines, etc. Finally, instead of interpretation notes, findings directly annotated on the visualizations can help users visually determine whether the interpretations missed interesting observable trends; they can prod the system to explain these observations via further analyses or request statistical significance tests on annotated observations.

## 5 THE LIMITATION & CHALLENGES

*Benchmarking.* The main hurdle to advancing any end-to-end automated system is creating a systematic and robust way of evaluating how well such a system works. A benchmark can guide implementation choices (e.g. prompting & prompt-engineering techniques, generating code vs in-LLM code execution, which LLM? etc.) and design choices (e.g. degree of human vs. cooperative AI-agent feedback, sequential prompting vs. all-in-one go). This benchmarking effort is non-trivial. In particular,

(1) *How do we assess value?* Defining rubrics such as relevance, viability, etc allow us to get an empirical handle on performance. However, what are the right rubrics and how do we avoid the trap of optimizing for what we can easily measure rather than what is useful? To illustrate, recent works that evaluate LLMs on creative writing tasks follow the approach of designing rubrics with the aid of experts and evaluating how well generated stories meet those rubrics [6, 12]. Yet the authors elaborate on how limited in scope these rubrics are and more importantly how they do not get at the question of how writers will make valuable use of generated stories.

(2) *How do we measure critical coverage?* For small datasets, one can imagine constructing a test set of critical analyses. Even then, the size of the set can be astronomical if one considers analysis that examines sub-sets or projections of the data. With the typical multi-dimensional data-sets of today, how do we know that an automated system is not missing on what we don't know? More nuanced than simply 'coverage' we need to balance how much we generate with how much of it is critical — did we miss out on a finding that can lead an investor to make poor investment choices?

(3) *How do we verify correctness?* This is a challenge of labor-intensity. For each generated visualization, the gold standard may be a human-constructed pipeline that attempts to study the same phenomena. Even then, how do we account for subtle differences in handling missing values, coding images, analyzing text, visual encoding choices, etc? As we discuss in §2 guardrails and ensuring correctness are critical especially if such a tool is to empower non-technical users, who do not have the skill-set to verify the findings.

(4) *How do we handle noise?* Even accounting for differences across generative runs, we note that slight backend modifications to third-party models can create wildly different results. In this environment, how do we create reproducible benchmarks?

## 6 RELATED WORK

Wu et al. provides an extensive survey of research on AI for visualization [21]. Automatic visualization tools such as VizDeck [13] and Draco [14] create the most *effective* visualization given a view or initial chart and can in essence replace or enhance our visualization step; they do not automate the end-to-end visual analysis process. Recent research has examined the power of LLMs in coding or the thematic analysis of multimodal data [7–9, 23, 24]: These works motivated our design choice to not distinguish between data and utilize LLMs to ingest such data.

SeeDB [19] defines *interesting* visualizations as those with large deviations from some reference or across subgroups within the data. To generate interesting visualizations, they develop heuristics to prune the combinatorial search space and scalably evaluate *intrestingness* across multiple possible visualizations. We see SeeDB as a complementary approach to ours; we use LLMs to *forward* confine from a narrative stakeholder perspective the space of attributes and metrics to visualize, while SeeDB *backward* narrows the search to those visualizations that are likely to show deviations.

Tools like Tableau's Einstein Copilot [4] and Open AI's GPT Data Analyst [2] provide a natural language interface for analysis and visualization. When appropriately prompted, these interfaces can respond to many data queries and can even generate reliable visualizations. However, users have to engineer their prompts; simply declaring one's interests does not lead to end-to-end analysis: GPT data analyst simply describes back the data sets and its schema. LLM4Vis's breakdown of the entire data analysis pipeline into sequential steps starting with *goal-setting* is novel.

## 7 CONCLUSION

With as little information as possible about a stakeholder, LLM4Vis uses LLMs to set analysis goals for the stakeholder, construct data processing scripts and generate visualization charts along with interpretations. There is a lot of room for improvement such as integrating human or even AI feedback, and much to be implemented such as guardrails. However, without a *robust* benchmark that assesses *value*, *critical coverage* and *correctness*, automated end-to-end analysis pipelines for the masses may very well be a pipe dream.

**Acknowledgments** This work was funded by Tamkeen, NYUAD Research Institute Award CG001 through NYUAD CITIES.

# REFERENCES


[1] 2024. ChatGPT Assistants API. https://platform.openai.com/docs/assistants/overview?context=with-streaming Accessed on March 23rd, 2024.

[2] 2024. GPT Data Analyst. https://chatgpt.com/g/g-HMNcP6w7d-data-analyst?oai-dm=1 Accessed on May 27th, 2024.

[3] 2024. Inside Airbnb. http://insideairbnb.com/ Accessed on February 27th, 2024.

[4] 2024. Tableau AI. https://www.tableau.com/products/tableau-ai Accessed on May 28th, 2024.

[5] Elif Akata, Lion Schulz, Julian Coda-Forno, Seong Joon Oh, Matthias Bethge, and Eric Schulz. 2023. Playing repeated games with Large Language Models. arXiv:2305.16867 [cs.CL]

[6] Tuhin Chakrabarty, Philippe Laban, Divyansh Agarwal, Smaranda Muresan, and Chien-Sheng Wu. 2024. Art or Artifice? Large Language Models and the False Promise of Creativity. arXiv:2309.14556 [cs.CL]

[7] Shih-Chieh Dai, Aiping Xiong, and Lun-Wei Ku. 2023. LLM-in-the-loop: Leveraging Large Language Model for Thematic Analysis. arXiv:2310.15100 [cs.CL]

[8] Jakub Drápal, Hannes Westermann, and Jaromir Savelka. 2023. Using Large Language Models to Support Thematic Analysis in Empirical Legal Studies. arXiv:2310.18729 [cs.AI]

[9] Cody Dunne, Carl Skelton, Sara Diamond, Isabel Meirelles, and Mauro Martino. 2016. Quantitative, Qualitative, and Historical Urban Data Visualization Tools for Professionals and Stakeholders. In *Distributed, Ambient and Pervasive Interactions*, Norbert Streitz and Panos Markopoulos (Eds.). Springer International Publishing, Cham, 405–416.

[10] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-Augmented Generation for Large Language Models: A Survey. arXiv:2312.10997 [cs.CL]

[11] Nahum Gershon and Ward Page. 2001. What storytelling can do for information visualization. *Commun. ACM* 44, 8 (aug 2001), 31–37. https://doi.org/10.1145/381641.381653

[12] Carlos Gómez-Rodríguez and Paul Williams. 2023. A Confederacy of Models: a Comprehensive Evaluation of LLMs on Creative Writing. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 14504–14528. https://doi.org/10.18653/v1/2023.findings-emnlp.966

[13] Alicia Key, Bill Howe, Daniel Perry, and Cecilia Aragon. 2012. VizDeck: self-organizing dashboards for visual analytics. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data* (Scottsdale, Arizona, USA) *(SIGMOD '12)*. Association for Computing Machinery, New York, NY, USA, 681–684. https://doi.org/10.1145/2213836.2213931

[14] Dominik Moritz, Chenglong Wang, Greg L. Nelson, Halden Lin, Adam M. Smith, Bill Howe, and Jeffrey Heer. 2019. Formalizing Visualization Design Knowledge as Constraints: Actionable and Extensible Models in Draco. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 438–448. https://doi.org/10.1109/TVCG.2018.2865240

[15] Arvind Satyanarayan and Jeffrey Heer. 2014. Authoring Narrative Visualizations with Ellipsis. *Computer Graphics Forum* 33, 3 (2014), 361–370. https://doi.org/10.1111/cgf.12392 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12392

[16] Edward Segel and Jeffrey Heer. 2010. Narrative Visualization: Telling Stories with Data. *IEEE Transactions on Visualization and Computer Graphics* 16, 6 (2010), 1139–1148. https://doi.org/10.1109/TVCG.2010.179

[17] Zeyuan Shang, Emanuel Zgraggen, Benedetto Buratti, Ferdinand Kossmann, Philipp Eichmann, Yeounoh Chung, Carsten Binnig, Eli Upfal, and Tim Kraska. 2019. Democratizing Data Science through Interactive Curation of ML Pipelines. In *Proceedings of the 2019 International Conference on Management of Data* (Amsterdam, Netherlands) *(SIGMOD '19)*. Association for Computing Machinery, New York, NY, USA, 1171–1188. https://doi.org/10.1145/3299869.3319863

[18] Evan R. Sparks, Ameet Talwalkar, Daniel Haas, Michael J. Franklin, Michael I. Jordan, and Tim Kraska. 2015. Automating model search for large scale machine learning. In *Proceedings of the Sixth ACM Symposium on Cloud Computing* (Kohala Coast, Hawaii) *(SoCC '15)*. Association for Computing Machinery, New York, NY, USA, 368–380. https://doi.org/10.1145/2806777.2806945

[19] Manasi Vartak, Sajjadur Rahman, Samuel Madden, Aditya Parameswaran, and Neoklis Polyzotis. 2015. *SeeDB: Efficient Data-Driven Visualization Recommendations to Support Visual Analytics*. Vol. 8. 2182–2193.

[20] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, Quoc Le, and Denny Zhou. 2022. Chain of Thought Prompting Elicits Reasoning in Large Language Models. *CoRR* abs/2201.11903 (2022). arXiv:2201.11903 https://arxiv.org/abs/2201.11903

[21] Aoyu Wu, Yun Wang, Xinhuan Shu, Dominik Moritz, Weiwei Cui, Haidong Zhang, Dongmei Zhang, and Huamin Qu. 2022. AI4VIS: Survey on Artificial Intelligence Approaches for Data Visualization. *IEEE Transactions on Visualization and Computer Graphics* 28, 12 (2022), 5049–5070. https://doi.org/10.1109/TVCG.2021.3099002

[22] Zhuofeng Wu, He Bai, Aonan Zhang, Jiatao Gu, VG Vinod Vydiswaran, Navdeep Jaitly, and Yizhe Zhang. 2024. Divide-or-Conquer? Which Part Should You Distill Your LLM? arXiv:2402.15000 [cs.CL]

[23] Xinlu Zhang, Yujie Lu, Weizhi Wang, An Yan, Jun Yan, Lianke Qin, Heng Wang, Xifeng Yan, William Yang Wang, and Linda Ruth Petzold. 2023. GPT-4V(ision) as a Generalist Evaluator for Vision-Language Tasks. arXiv:2311.01361 [cs.CV]

[24] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. arXiv:2306.05685 [cs.CL]