# A Survey of Visual Insight Mining: Connecting Data and Insights via Visualization

Yijie Lian[a], Jianing Hao[a], Wei Zeng[a,b], Qiong Luo[a,b]

[a] *The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China*
[b] *The Hong Kong University of Science and Technology, Hong Kong SAR, China*

---

**Abstract**

Insight mining transforms complex data into actionable knowledge, enabling effective decision-making across diverse domains. Given the richness and interpretative power of visualizations, visual insight mining - the process of extracting meaningful insights from raw data through intuitive visual representations - has become increasingly vital. This survey systematically reviews the current landscape of visual insight mining, addressing the critical questions: *"How can visualizations be generated from data?"* and *"How can insights be extracted from visualizations?"*. Specifically, we delve into six distinct tasks (*i.e.*, task decomposition, visualization generation, visualization recommendation, chart parsing, chart question answering, and insight generation) in the process of visual insight mining, and provide a comprehensive analysis of rule-based, learning-based, and large-model-based methods for each task. Based on the survey, we discuss current research challenges and outline future opportunities. By viewing visualization as a bridge in the data-to-insight path, this survey offers a structured foundation for further exploration in visual insight mining.

*Keywords:* Insight Mining, Data Visualization, Rule-based Methods, Learning-based Methods, Large-model-based Methods

---

## 1. Introduction

### *1.1. Background*

In the era of big data, insight mining that extracts meaningful insights from vast and complex datasets [1] has become a critical topic across various domains [2]. Despite its importance, this process has historically been complex and resource-intensive, requiring advanced expertise in statistical software, programming libraries, and domain knowledge [3, 4, 5, 6, 7, 8]. This expertise barrier creates an imbalance between data availability and actionable understanding - while organizations collect unprecedented amounts of information, its value remains locked without effective interpretation tools. In this situation, visualization [9] emerges as a powerful medium for bridging the gap between data and human understanding, enabling analysts to perceive patterns, identify anomalies, and communicate findings intuitively [10, 11, 12].

Visual insight mining is the process of using visualization to discover, extract, and communicate meaningful knowledge (such as patterns and trends) from complex data [13]. This approach leverages the power of visual representations to enhance human understanding and facilitate the discovery of significant insights that might not be apparent through traditional data analysis methods. In the context of visual insight mining, we mainly focus on automatically extracting insights from visualization, thereby reducing the burden on human analysts and enabling more rapid and accurate decision-making. Instead of focusing on how visualization designs may help humans extract insights directly from charts, visual insight mining emphasizes the automated presentation and identification of key information. The pipeline of visual insight mining generally starts with exploring statistical results from data [5], such as outstanding No.1, outlier, trend, correlation, and so on [14, 15]. Then, the statistical results are shown to users through visualizations, allowing them to discover meaningful insights intuitively. As illustrated in Figure 1, the process of visual insight mining can be generally divided into two stages of Data2Vis (*i.e.*, generating visualizations from data) and Vis2Insight (*i.e.*, deriving insights from visualizations). The methods used at each stage have undergone significant transformations, driven by advances in heuristic rules, machine learning (ML), and recently large models. However, little is known about these developments. This survey aims to fill the gap by addressing the following questions:

- **RQ1**: In the stage of Data2Vis, *"how can visualizations be generated from data?"*
- **RQ2**: In the stage of Vis2Insight, *"how can insights be extracted from visualizations?"*

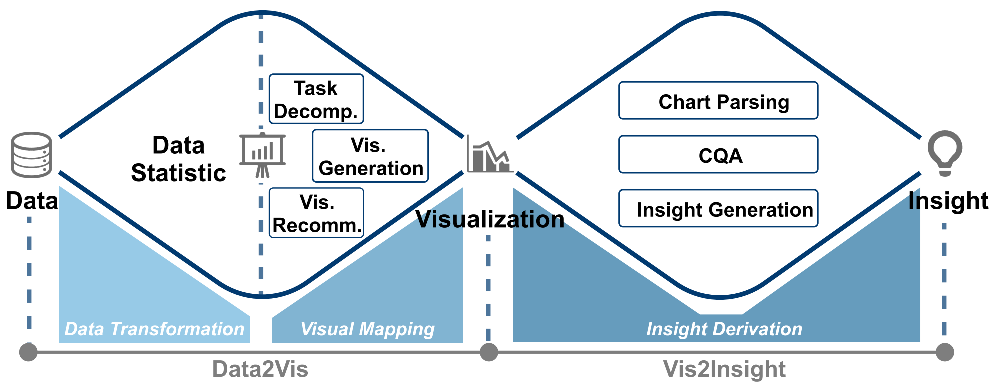


Figure 1: **Visual Insight Mining** extracts insights from data with visualization as the medium. The process consists of the transformation from data to visualization (Data2Vis), followed by automated or interactive insight extraction (Vis2Insight).

### *1.2. Method Landscape*

For **RQ1**, Data2Vis was conventionally a rule-based process that transforms structured data into visualizations through predefined logical rules. These rules typically mapped data attributes (e.g., numerical values, categories) to visual encodings (e.g., position, color, size) based on established principles from visualization theory [16, 17, 18, 19, 20, 21]. Rule-based systems relied on rigid templates and heuristics (e.g., automatically selecting scatter plots for correlations [22] or line charts for temporal trends). Through encoding visualization principles to optimize perceptual clarity, these systems effectively prune suboptimal designs, ensuring deterministic outputs. While effective for standardized reporting (e.g., business dashboards or scientific plots) with transparency and reproducibility, rule-based systems struggled to accommodate heterogeneous datasets, ambiguous user intents, and rigid rules [23, 5, 24]. This rigidity underscored the need for more adaptive and automated methods.

With the rise of statistical and ML [25], learning-based methods addressed some of the above limitations by embedding adaptive intelligence into the Data2Vis process [26]. For example, these methods leveraged knowledge graphs and sequence-to-sequence models to contextualize user requirements and generate visualization specifications [27, 28]. Moreover, researchers developed algorithms that could recommend optimal visual encodings based on dataset characteristics such as dimensionality, data types, and statistical properties [29]. However, the reliance on labeled training data limited their adaptability to evolving contexts and novel datasets [30, 27]. This limitation highlighted the importance of developing systems capable of generalizing across diverse analytical scenarios while minimizing dependence on extensive training resources.

Recently, the emergence of large language models (LLMs) [31, 32] has fundamentally transformed Data2Vis by enabling natural language (NL) interactions [33]. These models allow users to articulate analytical intentions in plain language (e.g., "Show sales trends by region") and then generate contextually appropriate visuals. Building on this foundation, advanced LLMs demonstrate greater adaptability, enabling users to focus more on hypothesis formulation and less on technical implementation [34, 35, 36, 37, 38]. By combining LLMs with Data2Vis, these tools enable users to move seamlessly from data to visualization, simplifying the visual design process and making it accessible to the general public.

For **RQ2**, the evolution of Vis2Insight follows a parallel trajectory. Early systems adopted rule-based methods grounded in established visualization theory and perceptual principles [20, 21, 22]. By encoding domain knowledge into deterministic rules, these systems could reliably identify fundamental chart components (*e.g.*, axes, legends) and detect statistical patterns (*e.g.*, correlations, outliers) through predefined analytical templates [39]. However, rule-based methods exhibited significant limitations when confronted with novel visualization formats or complex data patterns that exceeded the scope of predefined rules. These limitations became particularly apparent as visualization practices diversified across disciplines and data complexity increased in real-world analytical scenarios, constraining analytical flexibility and thus motivating the development of more adaptive methods.

As computer vision and ML techniques developed, learning-based methods emerged as a powerful tool for Vis2Insight applications. These data-driven methods overcame many limitations of rule-based systems by automatically learning to extract insights from visualization through exposure to annotated training examples. The field progressed significantly with the adoption of Deep Learning(DL) [40, 41] and Reinforcement Learning (RL) [42], achieving greater flexibility and broader applicability. However, learning-based methods introduced new challenges. For example, the performance of learning-based methods was found to be particularly dependent on the quality and diversity of annotated training data, with specialized architectures often required for different chart genres and analytical tasks [43, 44], highlighting the need for more generalizable and interpretable methods.

More recently, the development of multimodal large language models (MLLMs) [45, 46] provides a unified and powerful method to overcome the above challenges. These models integrate textual and visual reasoning capabilities, enabling holistic comprehension of complex datasets and visualizations. For example, MLLMs align textual and visual modalities to support a wide range of tasks, including anomaly detection, chart-based question answering, and hypothesis generation [47, 48, 49, 50, 51, 52, 53, 54, 55]. These systems not only answer questions but also proactively propose hypotheses, enabling collaboration between human and computer in Vis2Insight [55, 56].

Specifically, the integration of LLMs and MLLMs has simplified the entire visual insight mining pipeline. While LLMs facilitate transforming NL into code or visualization specifications [34, 36, 57], multimodal agents allow users to fluidly navigate between exploration, visualization, and interpretation [45, 58]. Emerging tools further enhance accessibility, enabling users to guide analysis through intuitive interactions like sketches or conversational agents [59]. As these technologies continue to evolve, they promise to transform visual insight mining into a universally accessible practice.

### *1.3. Paper Organization*

In this survey, we aim to investigate and report on different methodologies for mining insights through visualization. The remaining chapters of this survey are organized as follows: Chapter 2 introduces the definition of concepts, reviews related surveys, and presents the taxonomy of visual insight mining. Chapter 3 focuses on the Data2Vis stage of visual insight mining, which encompasses three key tasks: task decomposition, visualization generation, and visualization recommendation. We discuss the methodologies and techniques used in rule-based, learning-based, and large-model-based methods to effectively transform raw data into meaningful visualizations. Chapter 4 delves into the Vis2Insight stage of visual insight mining, which includes three key tasks: chart parsing, chart question answering, and insight generation. We explore the methodologies and techniques used in rule-based, learning-based, and large-model-based methods to extract actionable insights from visualizations. Chapter 5 discusses research challenges and opportunities in the field of visual insight mining, highlighting areas that require further exploration and development. Finally, Chapter 6 concludes the survey by emphasizing the crucial role of visualization as a medium between data and insights. By combining historical context with cutting-edge innovations, we aim to provide a comprehensive foundation for future research in visual insight mining.

## 2. Preliminaries

In this section, we lay the groundwork for our study by introducing key definitions, reviewing related surveys, and presenting a taxonomy of visual insight mining.

### *2.1. Definition of Concepts*

We begin with formally defining the three phases and six tasks that will be used throughout this paper.

#### *2.1.1. Three Phases of Visual Insight Mining*

In Figure 1, the visual insight mining pipeline bridges the gap between raw data and meaningful insight through intuitive visualization, encompassing three key phases: Data Transformation, Visual Mapping, and Insight Derivation. This pipeline transforms quantitative information into human-interpretable insights, with each stage influencing the effectiveness of subsequent steps. Beginning with raw data, the Data Transformation phase first transforms raw data into initial statistical results. The subsequent Visual Mapping phase generates visualization that effectively reveals patterns and relationships within statistical results. Finally, the Insight Derivation phase applies analytical techniques to extract meaningful knowledge from the visualizations. Notably, while the Data Transformation phase covers a substantial body of research, often involving statistical analysis and data preprocessing, our scope is dedicated to the *visual* aspects of insight mining. Consequently, our review primarily focuses on the processes of Visual Mapping and Insight Derivation, examining how visualizations are constructed and how insights are subsequently derived from them. Work within the Data Transformation phase falls outside the purview of this survey. Below are the definitions of three phases.

- ***Data Transformation*** is the process of converting raw data into a structured format, aiming to bridge the gap between initial data and the organized data statistic [60]. This conversion makes the data more suitable for accurate statistical analysis, enabling the derivation of meaningful insights and reliable calculations like averages, frequencies, and correlations.

- ***Visual Mapping*** [20] refers to the systematic transformation of data statistics into visualization such as charts, graphs, and dashboards.

The goal of visual mapping is to present complex data relationships and patterns in an intuitive manner, making them easier to understand and interpret [21]. Through visualization, analysts can more effectively identify outliers, trends, and correlations in the data.

- ***Insight Derivation*** involves the extraction of valuable insights and conclusions from visualization, which is guided by domain expertise and analytical reasoning [61]. Moving beyond data transformation, insight derivation addresses the "why" and "how" behind observed phenomena, tests hypotheses, and generalizes findings to broader contexts. The purpose of insight derivation is to discover the deeper meanings and business value behind the data by analyzing and interpreting the visualizations. These insights can help decision-makers make more informed decisions, optimize business processes, or uncover new business opportunities.

By leveraging human's superior pattern recognition capabilities, visualization enables more efficient discovery of meaningful patterns compared to numerical analysis alone. This perceptual advantage makes it essential for deriving meaningful insights from modern, multidimensional datasets. Nevertheless, the concept of visualization can be diverse. Below, we give the formal definition of visualization within the scope of this work.

***Visualization*** is the representation of information or data through common visual elements, such as charts, graphs, maps, and animations [9, 10, 62, 63], often enhanced with interactive capabilities to support dynamic exploration [64, 65].

As a critical component of visual insight mining, visualization enables data analysts to directly observe, interact with, and understand complex datasets more effectively than mere numerical representations [10, 66]. By presenting data visually, trends and outliers become apparent immediately, creating quicker understanding and more efficient communication of key findings. This aligns with Ben Shneiderman's principle that "The purpose of visualization is *insight*, not pictures."- emphasizing that visual representations are not final results, but powerful tools for insight mining [67]. Indeed, compared to conventional spreadsheet analysis, the cognitive efficiency of visual processing enables analysts to extract meaningful knowledge from multidimensional datasets with greater speed and accuracy! [68], significantly enhancing Insight Mining and ultimately bridging the gap between data and true understanding.

*2.1.2. SubTasks*

The Visual Mapping phase includes three key tasks: Task Decomposition, Visualization Generation, and Visualization Recommendation. Although there might be some internal overlap among these tasks, they are designed to progress linearly, each contributing to the next in a sequential manner: Task Decomposition helps understand user requirements, Visualization Generation creates charts based on these needs, and Visualization Recommendation further refines chart selection and design.

- ***Task Decomposition*** refers to the process of breaking down a complex task into smaller, more manageable subtasks [69, 70]. These subtasks are defined with specific objectives, inputs, and outputs, and they can be addressed individually. This process involves identifying the key components and steps required to achieve a larger goal, and organizing them in a way that facilitates more efficient and effective execution [71]. Task decomposition is necessary for simplifying the overall task and improving overall task performance.

- ***Visualization Generation*** is the process of creating visual representations of data to facilitate the understanding and communication of information. This process involves interpreting user inputs, which can be in the form of NL queries or other specifications, and transforming these inputs into appropriate visual encodings such as charts, graphs, and infographics [28, 35]. The goal is to accurately capture user intents and map them to suitable visualizations that effectively convey the underlying data patterns and insights [34, 38].

- ***Visualization Recommendation*** refers to the process of suggesting appropriate visual representations for data based on the characteristics of the data and the user's analysis goals [72]. This involves evaluating various visualization options and selecting those that best convey the information, highlight key insights, and support effective data exploration [29]. Visualization recommendation systems aim to reduce the effort required to manually specify visualizations [72], providing users with a range of suitable visualizations that enhance their understanding and analysis of the data [73].

The Insight Derivation phase involves Chart Parsing, Chart Question Answering (CQA), and Insight Generation. Similarly, while these tasks might have internal overlaps, our main consideration is this linear development, where each task sets the stage for the subsequent one, ensuring a coherent and structured approach from data visualization to the generation of insights.

- ***Chart Parsing*** refers to the process of extracting visual elements, textual information, and underlying data from a chart and converting them into structured data or text descriptions [74, 53]. This process usually involves identifying key components of the chart, such as graphical marks (e.g., bars in a bar chart, lines in a line chart, or slices in a pie chart), their properties (e.g., position, size, color), and associated text labels (e.g., titles, axis labels, legends) [75]. The goal of chart parsing is to convert the visual representation of data into a structured format that can be easily understood and processed by machines, facilitating downstream tasks such as data analysis, visualization redesign, natural language description generation, and knowledge graph construction [74, 76, 77].

- ***Chart Question Answering*** is a task that involves automatically extracting and interpreting key information from charts to answer questions posed in NL [43, 44, 78]. It requires the ability to comprehend complex NL queries, recognize and interpret visual elements within the charts, and perform logical and arithmetic reasoning to derive accurate answers [79]. The input typically consists of an image of a chart and a query, while the output is an answer string that addresses the query based on the information contained in the chart.

- ***Insight Generation*** refers to the process of uncovering meaningful, actionable information or patterns within visualization through various analytical methods and techniques [15, 61]. This process can be facilitated by systems and frameworks that automate and enhance the exploration of visualization, providing users with relevant and significant insights in an effective manner [15, 80].

### *2.2. Related Surveys*

The field of Visual Insight Mining has seen significant advancements through the integration of artificial intelligence (AI) and ML techniques. Recent surveys such as ML4VIS [60], AI4VIS [81], and GenAI4VIS [82] have systematically investigated the application of these techniques across various visualization processes.

ML4VIS [60] systematically explores how ML techniques can enhance different stages of the visualization pipeline, both the Data2Vis and Vis2Insight phases. For Data2Vis, the survey highlights key ML-assisted tasks, such as Data-VIS Mapping, which automates chart type selection and visual encoding. For Vis2Insight, it discusses processes like VIS Reading (extracting and interpreting visual content) and Insight Communication (embedding insights into visualizations). While the ML4VIS pipeline effectively bridges these stages, the survey offers a macro-level analysis of specific tasks without deeper exploration, such as chart question answering. Our work addresses this gap by providing a detailed classification and comprehensive discussion of this task.

AI4VIS [81] surveys the emerging field of AI4VIS, defining visualization data as digital representations in formats such as graphics, programs, or hybrids. Their work explores how AI techniques process, analyze, and generate such data through feature engineering, feature learning, and internal representations. The authors further split the process into seven core tasks. While their focus is primarily on visualization operations and targets, our work takes a broader perspective by examining the entire pipeline from data to insight through visualization.

Ye et al. [82] presented a comprehensive survey on the use of generative AI (GenAI) in visualization, categorizing generative methods by data structures (e.g., sequences, tables, graphs, and spatial formats) and aligning them with visualization-specific tasks including data enhancement, visual mapping, stylization, and interaction. Their work adapts the classical visualization pipeline to reflect the broader capabilities of GenAI, offering a structured perspective on how generative techniques contribute to various stages of the visualization process. Unlike their focus on GenAI, our survey examines a broader range of methods, including rule-based, learning-based, and LM-based approaches, rather than solely GenAI-driven techniques.

### *2.3. Taxonomy*

As Table 1 illustrates, in this work, we delve into two primary stages: Data2Vis and Vis2Insight in visual insight mining. The Data2Vis stage encompasses three key tasks: Task Decomposition, Visualization Generation, and Visualization Recommendation. Similarly, the Vis2Insight stage comprises three tasks: Chart Parsing, CQA, and Insight Generation. For each of these six tasks, we explore various approaches, including *rule-based*, *learning-based*, and *large model-based* , to achieve the desired outcomes. This taxonomy offers a clear and concise framework for understanding the diverse methods employed in various visual insight mining tasks.

Table 1: Taxonomy with examples.

| | **Data2Vis** | | | **Vis2Insight** | | |
|---|---|---|---|---|---|---|
| | Task Decomp. | Vis. Generation | Vis. Recomm. | Chart Parsing | CQA | Insight Generation |
| Rule based | NL4DV [69] | NL4DV [69], Calliope [83], AutoProfiler [84], Diff in the loop [85] | Mackinlay [23], Voyager [86], CompassQL [87], Seedb [72], TaskVis [88] | Poco et al. [89], ChartKG [77] | Kim et al. [90], Hoque et al. [78] | Foresight [39], Zeng et al. [91], Vertsel et al. [92] |
| Learning based | Talk2Data [93] | Data2Vis [28], ADVISor [94], DashBot [30] | DeepEye [95], VizML [29], Data2Vis [28], ML4VIS [60], KG4Vis [27] | ReVision [76], ChartSense [75], AutoCaption [74], Chart-to-Text [53], ChartReader [96], ChartKG [77] | Kafle et al. [97], PlotQA [43], ChartQA [44] | InkSight [59], Kim et al. [90], DashBot [30], Foresight [39] |
| LM based | AI Chains [70], LightVA [71], FinFlier [98], DashChat [99] | Chat2VIS [34], LIDA [35], DashChat [99], ChartGPT [38], Li et al. [100] | LLM4Vis [36], DracoGPT [101], Prompt4Vis [102] | Liew et al. [103], Pix2Struct [104], MATCHA [79] | Zeng et al. [105], MATCHA [79], AskChart [47], ChartBench [49], ChartAssistant [55] | InkSight [59], AutoTitle [106], VisTR [107], LEVA [61], InsightLens [80], InsightPilot [15], ChartInsighter [54] |

## 3. Data2Vis

Data2Vis focuses on the automated transformation of raw data into meaningful visual representations, which is a critical process in visual insight mining. This section is structured into three key subsections, including Task Decomposition, Visualization Generation, and Visualization Recommendation, to systematically review how related systems can facilitate converting data into effective visualizations.

### 3.1. Task Decomposition

Task decomposition transforms complex queries into simpler, actionable tasks to enhance problem-solving efficiency, building foundations for future steps. Various approaches have been explored to achieve this, including rule-based, learning-based, and large-model-based methods, each offering unique advantages and challenges.

#### 3.1.1. Rule-based Approaches for Task Decomposition

Rule-based methods for task decomposition involve using predefined rules and heuristics to break down complex tasks into simpler subtasks. These methods are deterministic and rely on explicit, handcrafted rules that capture the structure and semantics of the tasks. Rule-based methods are particularly useful when the task structure is well-understood and can be explicitly defined. They are often preferred for their transparency and interpretability, as the rules can be easily understood and modified by humans.

NL4DV [69] employs predefined rules and patterns to interpret NL queries for data visualization, breaking down the queries into components such as data attributes, analytic tasks, and visualization types. For example, it identifies keywords in the query that correspond to specific tasks like "correlation" or "distribution". This structured decomposition allows the system to further generate appropriate visualizations based on the inferred tasks and attributes from the query.

#### 3.1.2. Learning-based Methods for Task Decomposition

Learning-based methods for task decomposition leverage the power of ML or DL to automatically identify and separate tasks into subtasks, often optimizing for specific objectives such as efficiency, accuracy, or adaptability.

To address the challenge of complex questions in exploratory visual analysis, Talk2Data [93] proposes a novel approach that leverages DL to decompose intricate queries into simpler, more manageable sub-questions. It extends the classic sequence-to-sequence architecture with a question type classifier, decomposition layer, attention mechanism, and copying mechanism as shown in Figure 2. This allows the system to effectively handle complex queries by simplifying them into a series of straightforward tasks, which can be individually addressed.

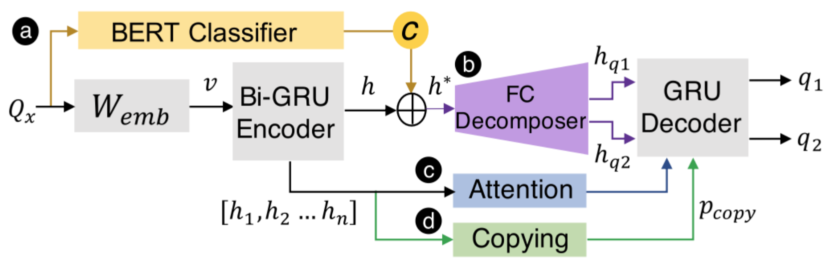


Figure 2: The decomposition mechanism of Talk2Data [93]. (Extracted from Talk2Data [93])

*3.1.3. Large-model-based Techniques for Task Decomposition*

Recent advancements in LLMs have significantly enhanced the capabilities of AI systems to tackle complex tasks through task decomposition. These methods leverage the inherent reasoning and generative abilities of LLMs to break down intricate problems into manageable subtasks, thereby improving the efficiency and effectiveness of problem-solving processes.

Wu et al. [70] introduced the concept of "Chaining", where complex tasks are decomposed into a series of smaller, more manageable subtasks that are processed sequentially by LLMs. In the case of visualization code debugging, each step in the Chain focuses on a specific subtask, such as rewriting Vega-Lite specifications into NL, validating design constraints, and generating fixes. This modular approach not only improves the quality and transparency of the visualization workflow but also enhances the model's controllability and debuggability, making it easier to identify and correct errors in the visualization workflow.

As shown in Figure 3, LightVA [71] presents a lightweight visual analytics (VA) framework that employs LLM agents for task planning and execution. It uses a recursive process with a planner, executor, and controller to dynamically handle task complexity. The planner decomposes tasks into smaller subtasks, the executor manages task execution (including visualization generation and data analysis), and the controller oversees the entire process. This decomposition approach makes the development and use of VA systems more controllable and transparent.

FinFlier [98] breaks down the complex task of generating graphical overlays for financial visualizations into two main subtasks: text-data binding and graphics overlaying. The text-data binding module uses advanced prompt engineering techniques (such as output constraint, CoT, and dynamic prompt)

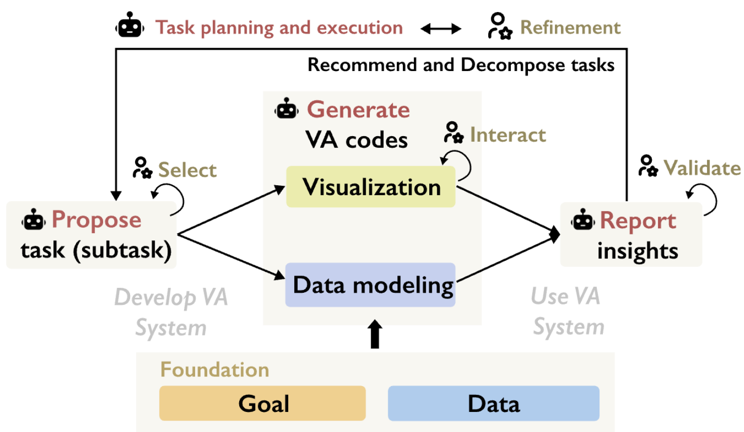


Figure 3: The conceptual framework of LightVA [71]. (Extracted from LightVA [71])

and a financial domain knowledge-grounded LLM to establish a robust connection between financial vocabulary in the text and data points in the table. The graphics overlaying module then generates effective graphical overlays based on the output from the text-data binding module, considering narrative sequencing and the correspondence between graphical overlays and financial narratives. This decomposition approach enables FinFlier to efficiently handle complex financial narratives and data, improving the efficiency and accuracy of generating graphical overlays for financial visualizations.

DashChat [99] system utilizes a powerful LLM to transform users' NL inputs into specific visualization tasks. By employing a JSON-style domain-specific language, the system ensures that the LLM's outputs are structured and aligned with user requirements, effectively breaking down vague descriptions into actionable tasks for subsequent visualization generation. This approach not only enhances the accuracy of task interpretation but also improves the overall efficiency of creating dashboard prototypes.

In summary, large-model-based techniques for task decomposition have made significant progress in enabling AI systems to handle complex tasks. Each of them offers unique approaches to breaking down tasks into manageable subtasks, thereby improving the transparency, controllability, and efficiency of AI-driven problem-solving.

*3.2. Visualization Generation*

Visualization Generation explores techniques for automatically creating visualizations based on structured tasks, leveraging computational and design principles.

*3.2.1. Rule-based Approaches for Visualization Generation*

Rule-based approaches for visualization generation rely on predefined rules and guidelines to automatically create visual representations from data. These methods are designed to map data attributes and user tasks to appropriate visualization types and encodings based on established visualization principles. By following these rules, visualization systems can generate effective and contextually relevant visualizations without requiring extensive user input or customization. This approach is particularly useful in scenarios where rapid and accurate visualization generation is needed.

NL4DV [69] employs rule-based methods to map detected data attributes and tasks inferred from NL queries to relevant visualizations. Specifically, the toolkit uses predefined rules to determine the most suitable visualization types and encodings based on the identified elements. For example, if the query specifies a correlation task between two quantitative attributes, NL4DV's map it to scatterplots. These rules ensure that NL4DV generates visualizations aligned with both the data characteristics and the user's intent expressed in the query.

Several systems investigate how to automatically generate visual representations of data based on predefined rules. Specifically, Calliope [83] introduces a system that automatically generates visual data stories from spreadsheets. The system defines a set of data fact types and corresponding visualization rules. These rules are based on empirical data and design guidelines to ensure that the visualizations are effective and easy to understand. For instance, a "trend" fact type is visualized using a line chart, while a "distribution" fact type is represented by a bar chart. These rules allow Calliope to automatically generate coherent and meaningful data stories, where each one is accompanied by an appropriately chosen visualization. Similarly, AutoProfiler [84] aims to generate visual summaries of data. The system automatically detects the data type of each column (e.g., numeric, categorical, temporal) and applies predefined rules to visualize the data distribution and summary statistics, so that data scientists can quickly understand their data through visualization.

In contrast, Diff in the loop [85] introduces a method for visualizing differences in datasets during exploratory data analysis. The system identifies correspondences between data points in the original and modified datasets and applies predefined rules to render these differences in various visualization views, such as parallel view, opacity view, and delta view. These views are designed to highlight changes in data distributions and summary statistics, providing data scientists with a clear understanding of the impact of their code modifications on the data.

#### *3.2.2. Learning-based Methods for Visualization Generation*

Learning-based methods for visualization generation leverage ML models to capture patterns and relationships within datasets and automatically generate visualizations that are both meaningful and contextually relevant.

Building on the foundation of DeepEye, subsequent research has explored more sophisticated ML models to enhance visualization generation. For example, Data2Vis [28] formulates visualization design as a language translation problem, where data specifications are mapped to visualization specifications in a declarative language (Vega-Lite). As shown in Figure 5, Data2Vis trains a multilayered attention-based encoder-decoder network with long short-term memory (LSTM) units on a corpus of visualization specifications. This approach allows the model to learn the vocabulary and syntax for valid visualization specifications, appropriate transformations, and common data selection patterns. Data2Vis demonstrates the efficacy of bidirectional models with attention mechanisms for generating visualizations that are comparable to manually created ones in a fraction of the time.

ADVISor [94] leverages DL models to interpret NL questions and generate corresponding visualizations. Specifically, it utilizes a pre-trained language model like BERT to encode the semantics of the questions and table headers, which helps in understanding the user's intent and selecting the relevant data attributes. Based on the parsed information, the system dynamically chooses appropriate visualization types (e.g., bar charts, line charts, scatter plots) and generates visualizations with annotations to highlight the answers. This approach enables the system to handle a wide range of NL queries and produce meaningful visualizations that aid in data exploration and analysis.

Further, DashBot [30] introduces a deep reinforcement learning framework to generate analytical dashboards, addressing the complexity of creating effective dashboards that require both data analysis expertise and familiarity with professional tools. The model formulates dashboard generation as a Markov decision process, where agents explore the design space by taking actions such as adding or removing charts and configuring chart parameters. DashBot uses an asynchronous advantage actor-critic (A3C) algorithm to train agents, incorporating reward functions that evaluate the expressiveness, insightfulness, and diversity of generated dashboards. Its effectiveness is demonstrated through ablation studies and user studies, showing its ability to generate high-quality dashboards that are understandable, insightful, and aesthetically pleasing.

*3.2.3. Large-model-based Techniques for Visualization Generation*

With the development of LLMs, the field of visualization generation has seen significant advancements. These advancements leverage the strengths of LLMs to bridge the gap between natural language and data visualization, making the creation of insightful visualizations more accessible and efficient for a broader audience.

Chat2VIS [34] marks the initial efforts in this domain, which introduces the use of LLMs like GPT-3 and Codex to directly convert NL queries into executable Python code for generating visualizations. This approach not only streamlines the process but also reduces the complexity and cost associated with developing natural language interfaces (NLIs) for visualization.

Building on this foundation, LIDA [35] presents a more comprehensive tool that frames visualization generation as a multi-stage task. By integrating LLMs with image generation models, LIDA expands the scope of automated visualization creation to include both charts and infographics. Its modular design, consisting of SUMMARIZER, GOAL EXPLORER, VISGENERATOR, and INFOGRAPHER, enables a more structured way to generate visual representations. Similarly, DashChat [99] leverages LLMs to create industrial dashboard prototypes. After decomposing user requirements into specific tasks, it uses LLMs to select appropriate chart types (mapping visualization fields such as axes and color encodings) and simulate data as realistic as possible, ensuring that the visualizations are meaningful and aligned with user intents. The integration of LLMs enables DashChat to produce high-quality, contextually appropriate visualizations efficiently, even without access to real datasets.

To address the challenges of specifying complex parameters, ChartGPT [38] further decomposes the visualization generation process into a step-by-step reasoning pipeline (as shown in Figure 4). This solution enables ChartGPT to focus on solving one specific step at a time, thereby improving the accuracy and reliability of the generated visualizations. ChartGPT's innovation also includes the creation of a dataset of abstract utterances and correspond-

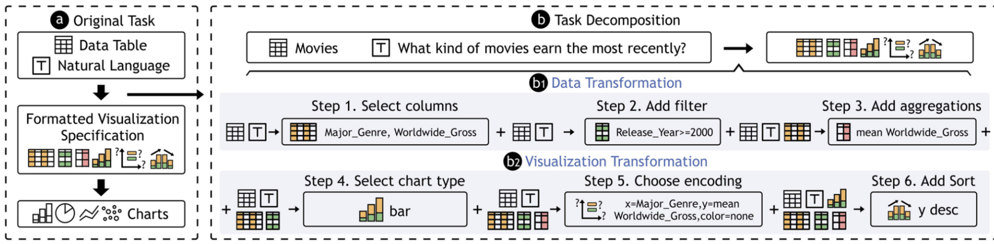


Figure 4: ChartGPT [38] decomposes the visualization generation process into three steps (4, 5, and 6). (Extracted from ChartGPT [38])

ing charts. The dataset is used to fine-tune the LLM, further enhancing its performance in generating visualizations from ambiguous or underspecified NL inputs.

More recently, Li et al. [100] provided deeper insights into the capabilities and limitations of LLMs in generating visualization specifications. Their evaluation using GPT-3.5 on the nvBench dataset demonstrates that few-shot prompting strategies significantly outperform zero-shot strategies, highlighting the importance of providing examples to guide the model. This study also underscores the need for LLMs to understand Vega-Lite grammar and task descriptions better, suggesting future directions for improving both the models and the benchmark datasets.

### *3.3. Visualization Recommendation*

Visualization Recommendation discusses intelligent systems that suggest optimal visualizations by considering data attributes, user preferences, and contextual constraints.

#### *3.3.1. Rule-based Approaches for Visualization Recommendation*

Rule-based methods for visualization recommendation have evolved significantly over the years, driven by the need to automate the process of generating effective and insightful visualizations. These methods rely on predefined rules and constraints to rank and select visualizations based on various factors such as data types, user tasks, and perceptual effectiveness.

Mackinlay [23] proposed APT (A Presentation Tool), which serves as the foundation of rule-based visualization recommendation. APT uses a set of predefined rules to prune and rank visual encodings based on their expressiveness and effectiveness. This approach ensures that the generated visualizations convey all the facts in the data and are easily perceived by users. Moreover, APT is notable for its flexibility and adaptability, allowing designers to incorporate new rules and refine existing ones to improve the quality of recommended visualizations.

Building on APT, subsequent systems further refine the rule-based approaches. For instance, Voyager [86] introduces a mixed-initiative system that combines user input with automated recommendations, enabling users to explore different visualizations through faceted browsing. Its recommendation engine, Compass, uses a set of rules to generate and rank visualizations based on data properties and user tasks. In this way, Voyager not only enhances the relevance of recommended visualizations but also provides users with more control over the recommendation process.

CompassQL [87], another significant development in rule-based approaches, formalizes visualization design knowledge as constraints and uses Answer Set Programming (ASP) to solve these constraints. This approach allows for a more systematic and extensible representation of design knowledge, enabling the integration of various empirical studies and best practices. CompassQL's ability to learn weights for soft constraints from experimental data further enhances its flexibility and adaptability, allowing it to incorporate new findings and preferences into the recommendation process.

Seedb [72] represents a practical application of rule-based methods in the context of data-driven visualization recommendations. Seedb uses a deviation-based metric to evaluate the utility of visualizations, identifying those that show significant deviations from a reference dataset as potentially interesting. This metric is encoded as a rule within the system, guiding the selection of visualizations that highlight notable trends or anomalies in the data. Seedb's approach is particularly effective in high-dimensional datasets where manual exploration is impractical, making it a valuable tool for rapid visual analysis.

TaskVis [88] shows a more recent advancement in rule-based visualization recommendation. TaskVis focuses on task-oriented visualization recommendations by modeling user analysis tasks in detail. It maintains a task base that includes 18 classical low-level analysis tasks and their appropriate chart types, and uses a rule base to extend empirical wisdom with targeted modeling of analysis tasks. Further, TaskVis employs ASP to enumerate candidate visualizations and supports multiple ranking schemes based on chart complexity, user interest, and task coverage. This approach ensures that the recommended visualizations are not only effective but also closely aligned

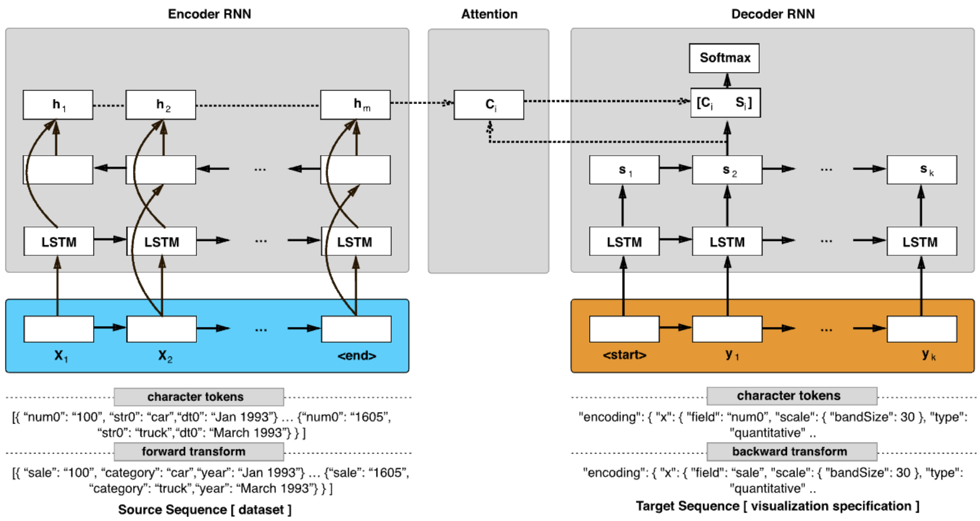


Figure 5: Data2Vis [28] employs a sequence-to-sequence model that translates data specifications into visualization specifications. (Extracted from Data2Vis [28])

with the user's specific analysis tasks.

### *3.3.2. Learning-based Methods for Visualization Recommendation*

Learning-based methods leverage ML models to learn visualization design principles directly from data, offering a more scalable and adaptable solution.

DeepEye [95] introduces a novel system for automatic data visualization that uses learning-based methods to recommend visualizations. It addresses three key problems: recognizing whether a visualization is good or bad using a binary classifier, ranking visualizations using a supervised learning-to-rank model, and selecting the top-k visualizations from a dataset. These learning-based methods help capture human perception of what makes a visualization effective and enable the system to automatically recommend compelling visualizations for a given dataset.

VizML [29] leverages a large corpus of datasets and associated visualizations to train ML models for visualization recommendation. The system identifies key design choices made during visualization creation, such as selecting visualization types and encoding data columns along specific axes. Then, it employs neural networks to predict these design choices, achieving high accuracy compared to baseline models. VizML also evaluates feature importances, providing insights into the contribution of different data features to visualization effectiveness. The model's generalizability is assessed through crowdsourced benchmarking, where it performs comparably to human users and outperforms other visualization recommendation systems.

Data2Vis [28] proposes another innovative method to effectively recommend visualizations, which translates data specifications into visualization specifications using a sequence-to-sequence recurrent neural network model, as depicted in Figure 5. Trained on a corpus of Vega-Lite visualization specifications, the model learns common patterns and best practices in visualization design, which it uses to recommend the most suitable visualizations for a given dataset. By learning the vocabulary and syntax of valid visualization specifications, the model suggests visualizations that are not only syntactically correct but also semantically meaningful.

ML4VIS [60] provides a comprehensive overview of how ML techniques are being utilized to enhance visualization recommendations. The survey emphasizes the importance of ML in automating visualization recommendation, which can help users, especially those without extensive visualization expertise, to quickly identify the most appropriate visual representations for their data. It highlights that ML models can learn from large datasets of existing visualizations to understand patterns and best practices, thereby enabling the recommendation of suitable visualization types and configurations for new datasets. Additionally, the paper discusses the potential of ML to personalize visualization recommendations based on user preferences and interaction history, further enhancing the user experience. By mapping out the current landscape of ML applications in visualization recommendation, the survey offers valuable insights for researchers and practitioners looking to leverage ML for more effective and user-friendly visualization tools.

Further, KG4Vis [27] proposes a knowledge graph (KG)-based method for visualization recommendation, addressing the limitations of traditional ML approaches. It constructs a KG comprising entities such as data features, data columns, and visualization design choices, and the relationships between them. The model uses TransE-based embedding techniques to learn the embeddings of entities and relations from existing dataset-visualization pairs. These embeddings capture the underlying visualization rules, enabling the model to infer effective visualizations for new datasets. KG4Vis not only recommends visualizations but also generates explainable rules, enhancing user trust and understanding. The approach is evaluated through quantitative comparisons, case studies, and expert interviews, demonstrating its effectiveness and interpretability.

*3.3.3. Large-model-based Techniques for Visualization Recommendation*

With the advent of LLMs like ChatGPT [108, 109] and GPT-4 [31, 109, 33, 110], large-model-based methods for visualization recommendation have shown substantial progress, leveraging the capabilities of LLMs to provide more intuitive, accurate, and user-friendly visualization suggestions. These models leverage in-context learning to perform complex tasks without extensive supervised training, making them promising candidates for visualization recommendation.

LLM4Vis [36] focuses on using LLMs to generate visualization recommendations directly from NL queries. This approach eliminates the need for users to have in-depth knowledge of visualization tools and languages. Specifically, LLM4Vis provides an NLI for users to describe their visualization needs, in which LLMs process these queries and translate them into visualization specifications. Further, by leveraging the vast knowledge and training data of LLMs, LLM4Vis can generalize well across different datasets and user queries, offering robust and flexible visualization recommendations. Moreover, LLM4Vis supports interactive feedback, allowing users to refine their queries and receive updated recommendations iteratively.

To further enhance the capabilities of LLMs, DracoGPT [101] incorporates additional techniques such as the Draco framework [111] to provide higher-quality visualization suggestions. Draco, originally designed as a constraint-based system for automated visualization design, models visualization design knowledge using constraints and associated weights. DracoGPT extends this by incorporating the power of LLMs to generate more nuanced and contextually relevant visualization suggestions. This integration allows the system to interpret user queries more effectively and generate visualization recommendations that are not only syntactically correct but also semantically aligned with user intent. Additionally, DracoGPT learns preferences from experimental data, enabling it to adapt to different user needs and data characteristics.

Prompt4Vis [102] represents the latest advancement in large-model-based techniques for visualization recommendation. To improve the accuracy and relevance of visualization recommendations, Prompt4Vis introduces a novel framework that enhances the performance of LLMs in generating data visualizations by using example mining and schema filtering. The multi-objective example mining module selects the most effective examples for in-context learning, considering similarity, influence, and diversity to ensure that the LLM receives high-quality and relevant context. Additionally, the schema filtering module simplifies the database schema, reducing irrelevant information and improving the LLM's ability to generate more accurate visualizations. Extensive experiments demonstrate that Prompt4Vis significantly outperforms previous state-of-the-art methods, achieving substantial improvements in accuracy.

## 4. Vis2Insight

Vis2Insight explores the process of extracting meaningful knowledge from visualizations, bridging the gap between visual representation and human understanding. This section is organized into three core subsections, including Chart Parsing, Chart Question Answering, and Insight Generation, to systematically address how machines parse, interpret, and derive insights from visual data. Together, these subsections provide a comprehensive framework of how related systems transform visualizations into actionable insights, enabling deeper visual understanding.

### *4.1. Chart Parsing*

Chart Parsing examines techniques for decomposing visualizations into structured representations, further enabling information extraction.

#### *4.1.1. Rule-based Approaches for Chart Parsing*

Rule-based methods have been consistently applied in chart parsing to establish relationships between visual elements and data values, infer chart properties, and classify text roles. These methods provide a structured and interpretable approach to extracting meaningful information from chart images, enabling various downstream applications such as knowledge graph construction, data extraction, and visualization redesign.

Poco et al. [89] employed heuristic rules to enhance the accuracy and reliability of chart parsing. Specifically, a set of rules is applied to the output of SVM to correct and optimize the classification of text elements' roles, thereby significantly enhancing the accuracy of text role classification. Also, rules are employed to parse axis or legend label text to determine whether the data is quantitative or nominal, laying the groundwork for subsequent chart specification generation.

In ChartKG [77], predefined rules play a crucial role in constructing a structured knowledge graph of a chart. A total of four relationships are defined and constructed by rules to ensure that the knowledge graph can accurately capture the semantic information in the chart. Specifically, visual property correspondences are created by linking visual elements to their property values, such as connecting a bar to its height. Data variable correspondences are formed by associating data variable values with their corresponding variables. Visual encoding mappings are constructed by connecting visual element property values to data variables or values based on similarities, such as color or position. Visual insight correspondences are established by linking visual insights to the relevant data variables or values. These rules enable knowledge graphs to represent information in charts in a structured and interpretable way, thereby supporting various downstream tasks such as semantic-aware chart parsing and CQA.

#### *4.1.2. Learning-based Methods for Chart Parsing*

Learning-based methods have emerged as a powerful approach to address the challenges of chart parsing, leveraging ML and DL techniques to improve accuracy and efficiency. These methods have significantly advanced the field by automating the classification of chart types and extraction of visual elements from various types of charts.

ReVision [76] is a pioneering work that improves the automation of chart parsing and redesign through ML methods. Specifically, Revision is implemented in three stages. Firstly, Revision uses low-level image feature vectors for classification via support vector machines (SVM). It then uses a combination of image processing and model fitting techniques to locate graphical marks (visual elements that encode data) and extract underlying data tables from charts. Finally, based on the extracted data, ReVision automatically generates multiple redesigned charts, and users can select and compare different designs. With Revision, users can view the gallery of redesigned charts and accordingly change design according to specific visual aesthetics.

Building upon this foundation, with the advent of DL, more sophisticated methods have been proposed to handle the complexities of chart parsing. One notable example is ChartSense [75], which integrates deep learning-based classification with interactive data extraction algorithms. ChartSense first classifies chart types based on convolutional neural network (CNN), and then applies semiautomatic, interactive extraction algorithms optimized for each chart type. This mixed-initiative approach combines the strengths of automatic mark extraction algorithms and user interactions to improve the accuracy and efficiency of data extraction.

Both AutoCaption [74] and Chart-to-Text [53] leverage DL to generate textual descriptions of charts, but they differ in their focus and approach. While AutoCaption emphasizes the rapid generation of captions based on visual elements and their relationships, Chart-to-Text focuses on generating detailed summaries that capture the semantic content of the chart data. This distinction highlights the versatility of learning-based methods in addressing different aspects of chart parsing and user needs.

Most recently, researchers have aimed to develop unified frameworks that can handle multiple chart types and tasks. For example, ChartReader [96] integrates chart derendering and comprehension tasks using a transformer-based chart component detection module and a pre-trained vision-language model. The module can automatically learn chart rules from annotated datasets, eliminating the need for manual rule-making. Specifically, the Hourglass network is used to detect the center points and key points of chart components and group them through a multi-head attention mechanism. This approach enables the system to automatically identify and locate various elements in a chart without relying on predefined rules. By employing learning-based methods, ChartReader not only improves the accuracy of chart parsing but also reduces the manual effort involved in chart analysis.

ChartKG [77] represents the latest advancement in this domain, applying DL models to extract data from charts effectively before constructing a knowledge graph. It first uses ResNet50 to classify the input chart and determine the chart type (such as bar chart, line chart, etc.). For chart parsing, ChartKG applies YOLOv5 - a well-known object detection model with remarkable accuracies and scalabilities - to detect elements in the chart (such as bars in a bar chart, lines in a line chart, etc.). These deep learning-based methods allow ChartKG to accurately identify and extract various elements and their properties from different types of charts, providing a comprehensive and structured representation of the chart's visual and semantic information for subsequent knowledge graph construction.

#### *4.1.3. Large-model-based Techniques for Chart Parsing*

With the advent of LLMs and vision-language models, chart parsing, which involves extracting meaningful information from visual representations such as bar charts, line charts, and pie charts, has seen significant advancements. These models leverage the power of large-scale pretraining to improve the accuracy and generalizability of chart parsing tasks.

The journey begins with foundational work on generating NL summaries from charts. Chart-to-text [53] introduces a comprehensive benchmark to evaluate the performance of neural models in this domain. This work highlights the potential of models like Chart2text and T5 in generating fluent summaries, while also identifying key challenges such as factual errors and hallucinations. This benchmark serves as the foundation of future research by providing a large-scale dataset and setting the stage for more sophisticated models to build upon.

Then, Liew et al. [103] explored the capabilities of advanced language models like GPT-3 in capturing image captions. They primarily focused on prompt engineering, demonstrating that carefully crafted prompts can significantly enhance the quality and engagement of generated captions. This work underscores the importance of human interaction in guiding LLMs to produce more meaningful and contextually relevant outputs.

Pix2Struct [104] is one of the pioneering models in this domain. It introduces a novel pretraining strategy for visually-situated language tasks, focusing on parsing masked screenshots of web pages into simplified HTML. This approach allows the model to learn rich representations of textual and visual elements, which could then be transferred to various downstream tasks. Pix2Struct's architecture includes a variable-resolution input representation, which enables it to handle diverse input formats without distorting the original aspect ratio. This flexibility is crucial for tasks involving documents, illustrations, and user interfaces. In addition to its innovative pretraining objective, Pix2Struct demonstrates state-of-the-art (SoTA) performance on multiple benchmarks across different domains, setting a new standard for general-purpose visually-situated language understanding. Building on the foundation of Pix2Struct, MATCHA [79] incorporates the chart derendering task into the pretraining process, which involves generating the underlying data table or code used from a chart, improving the model's ability to extract numerical data from charts.

From the initial benchmarking and identification of challenges to the innovative use of prompt engineering, and culminating in the development of unified frameworks and specialized pretraining techniques, each step has built upon the previous one. This progression has not only improved the accuracy and efficiency of chart parsing but also paved the way for more universal and robust models capable of handling a wide range of chart-related tasks.

### 4.2. Chart Question Answering

Chart Question Answering is a complex task that involves interpreting visual data from charts and graphs to answer NL queries. This task is particularly challenging due to the need for both visual and linguistic understanding.

#### 4.2.1. Rule-based Approaches for Chart Question Answering

This transformation allows the system to leverage existing table question answering algorithms like Sempre [112, 113] to generate accurate answers.

Kim et al. [90] developed an automatic pipeline for answering questions about charts and generating visual explanations. The pipeline includes a stage where it converts visual questions into non-visual questions using rules. This involves detecting references to visual marks, attributes, and operations in the question and replacing them with references to data fields and values based on predefined rules and word lists. This rule-based conversion allows the system to transform questions that refer to visual features of the chart into a form that can be processed by ML algorithms.

Then, Hoque et al. [78] provided a comprehensive review of CQA systems. It mentions that early approaches to NLI for visualizations often rely on heuristic or grammar-based parsing techniques to handle questions about charts. These rule-based systems typically involve defining a set of predefined rules and patterns to match and interpret the NL queries. However, such methods have limitations in handling complex and diverse questions due to their reliance on predefined rules and lack of flexibility.

#### 4.2.2. Learning-based Methods for Chart Question Answering

Recent advancements in the field of CQA have been driven by the integration of computer vision and natural language processing techniques. These learning-based methods aim to automatically answer questions about data visualizations such as bar charts, line graphs, and pie charts.

Once data is extracted from charts, the next challenge is to effectively combine visual and linguistic features to answer questions. Kafle et al. [97] introduced PReFIL, a novel algorithm that learns bimodal embeddings by fusing question and image features. PReFIL then aggregates these embeddings to answer proposed questions. This approach significantly outperforms SoTA systems on the FigureQA [114] and DVQA [115] datasets, demonstrating the effectiveness of the proposed bimodal fusion technique.

Another significant challenge in CQA is dealing with out-of-vocabulary (OOV) words and complex reasoning tasks. To address this challenge, PlotQA [43] introduces a dataset with 28.9 million question-answer pairs over 224,377 plots, sourced from real-world data, and a hybrid model that combines visual perception, OCR, and table-based reasoning to handle both fixed-vocabulary and OOV questions. This model achieves SoTA results on both the DVQA and PlotQA datasets, highlighting the importance of integrating multiple techniques to handle diverse question types.

Instead of solely relying on charts, Masry et al. [116] performed CQA by combining automatic data extraction from chart images with SoTA table parsing models. Specifically, it uses computer vision techniques to recover the underlying data from chart images and then applies the TAPAS model, a BERT-based architecture, to parse the extracted data tables and generate answers to questions about the charts. Compared to methods that treat charts as regular images, this one leverages the unique structure of charts to improve the accuracy of question answering, demonstrating significant performance improvements on the FigureQA dataset.

Further, ChartQA [44] introduces a large-scale benchmark dataset for CQA that requires both visual and logical reasoning. It proposes two transformer-based models that combine visual features and the data table of the chart to answer questions. These models achieve SoTA results on previous datasets as well as on the new benchmark, demonstrating the potential of integrating visual and tabular data for complex reasoning tasks in CQA.

#### *4.2.3. Large-model-based Techniques for Chart Question Answering*

CQA is a critical task in the field of multimodal understanding, aiming to understand and reason about visual data to answer users' queries. With the advancements in LLMs and MLLMs, significant progress has been made in this area. Techniques such as visualization-referenced instruction tuning, math reasoning, and explicit integration of textual cues have been shown to enhance model performance.

At the beginning, MATCHA [79] presents a method that enhances visual-language models' capabilities in CQA through math reasoning pretraining task. MATCHA builds on the Pix2Struct model, a powerful image-to-text visual language model, and further pre-trains it with numerical reasoning task. By solving math problems rendered as images, MATCHA outperforms previous models on standard benchmarks like PlotQA and ChartQA, demonstrating its effectiveness in handling complex visual reasoning tasks.

In another attempt, AskChart [47] introduces a lightweight model that integrates both textual and visual cues from charts using a Mixture of Experts (MoE) architecture. AskChart explicitly incorporates textual information, such as data labels and axis labels, which are often overlooked by existing models. By aligning visual and textual modalities, AskChart achieves SoTA performance on multiple chart understanding tasks, including CQA. The model's effectiveness is further enhanced by a three-stage training strategy and a large-scale dataset, ChartBank, which contains over 7.5 million samples. This approach underscores the importance of leveraging both visual and textual information to achieve comprehensive CQA.

Table 2: Statistics of Benchmark Datasets.

| Datasets | #Chart type | #Task type | #Images | #Instruction data pairs |
|---|---|---|---|---|
| DVQA [115] | 1 | 3 | 300K | 3.4M |
| PlotQA [43] | 3 | 1 | 224K | 28M |
| ChartQA [44] | 3 | 1 | 21.9K | 32.7K |
| Chart-to-text [53] | 6 | 1 | 44K | 44K |
| Unichart [48] | 3 | 4 | 627K | 7M |
| StructChart [117] | 3 | 1 | 9K | 9K |
| ChartBench [49] | 9 | 5 | 66.6K | 599.6K |
| ChartX [51] | 18 | 7 | - | - |
| MMC [118] | 7 | 9 | 600K | 600K |
| ChartSFT [55] | 9 | 5 | 47K | 47K |
| ChartLlama [56] | 10 | 7 | 11K | 160K |
| NovaChart [119] | 18 | 15 | 47K | 856K |

In the meanwhile, the introduction of comprehensive benchmarks in Table 2 further highlights the need for models to effectively handle complex visual reasoning tasks. For instance, ChartBench [49] includes a wide variety of chart types and tasks, with a significant proportion of unannotated charts to assess models' ability to reason visually. The benchmark introduces an enhanced evaluation metric, Acc+, to provide more accurate and robust assessments. Extensive experiments on 21 mainstream MLLMs reveal their limitations in understanding complex charts, particularly those without data annotations. ChartBench also proposes two baselines based on chain of thought [120] and supervised fine-tuning [121] to improve model performance. Overall, future research should focus on developing more robust models that can generalize well to diverse chart types and tasks, ensuring reliable and accurate CQA in real-world applications.

Following this, ChartAssistant [55] introduces significant innovations. It

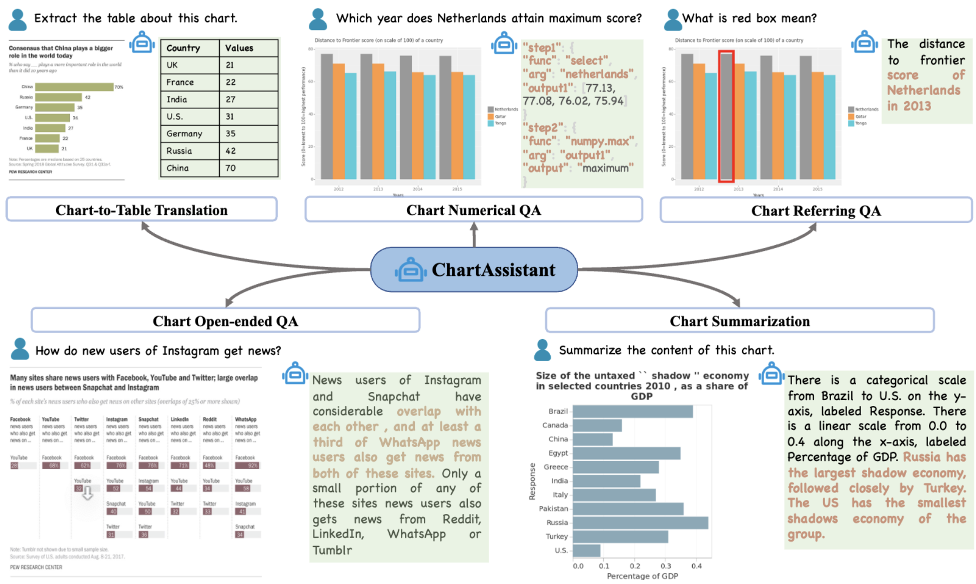


Figure 6: A range of chart understanding and reasoning tasks ChartAssistant [55] can perform. (Extracted from ChartAssistant [55])

employs a two-stage training process that begins with chart-to-table pre-training to align chart elements with structured text, followed by multitask instruction tuning on a comprehensive dataset called ChartSFT. This dataset encompasses a wide range of chart types and tasks, enabling ChartAssistant to achieve superior performance across various chart-related tasks, including CQA (as shown in Figure 6). By integrating chart-to-table translation and multitask instruction tuning, ChartAssistant not only improves its understanding of chart structures but also enhances its ability to perform mathematical reasoning and generate accurate answers to complex questions. This approach sets ChartAssistant apart from other large-model-based techniques, demonstrating its effectiveness in both generalization and task-specific performance, especially in zero-shot settings where it outperforms existing models significantly.

More recently, Zeng et al. [105] proposed a novel approach to enhance MLLMs for CQA by incorporating visualization-referenced instruction tuning. They identified limitations in existing datasets and models, such as biased chart distributions and inadequate fine-grained visual encodings. To address these issues, they introduced a data engine that filters and augments existing datasets, ensuring a more balanced and comprehensive representation of chart types and QA tasks. Their model, trained with this enriched dataset, demonstrates superior performance on established benchmarks, outperforming SoTA models with significantly fewer training examples.

### *4.3. Insight Generation*

Insight Generation investigates methods for automatically identifying and summarizing key patterns, trends, and anomalies from visual data.

#### *4.3.1. Rule-based Approaches for Insight Generation*

Rule-based approaches for insight generation leverage predefined rules and logic to extract meaningful insights from data. These approaches are particularly useful in scenarios where specific patterns or conditions need to be identified reliably and efficiently.

Foresight [39] provides a structured framework for exploring insights through predefined insight types and metrics. It categorizes insights into classes such as dispersion, skew, heavy tails, outliers, and linear relationships, each with associated ranking metrics and visualizations, allowing users to navigate the space of insights in a systematic manner. The predefined insight classes and metrics provided by Foresight serve as a foundation to guide the insight generation process.

In the domain of urban traffic analysis, Zeng et al. [91] employed the unit visualization techniques to enable transportation experts to systematically investigate how different spatial aggregations affect prediction accuracy. The system provides structured visual guidance for exploring the dynamic spatial variance in urban traffic and generating insights about spatial variations in prediction performance.

In the context of business intelligence, Vertsel et al. [92] explored the integration of rule-based systems with LLMs to enhance the extraction of actionable insights. The rule-based systems in this hybrid approach are designed to handle structured data with precision, ensuring that specific business rules and thresholds are met. This method allows for the identification of patterns and anomalies that align with predefined business logic, providing a reliable foundation for decision-making.

#### *4.3.2. Learning-based Methods for Insight Generation*

Learning-based methods for insight generation have achieved great advancements in recent years, driven by the need to automate and enhance the process of extracting meaningful insights from complex data.

InkSight [59] significantly enhances insight generation in computational notebooks by leveraging sketch interaction. It allows users to indicate data items of their interest through intuitive sketching on charts, which is then interpreted by the system using various learning-based methods. Specifically, it employs the Interquartile Range (IQR) method for outlier detection, linear regression for trend identification, and the Pearson Correlation Coefficient for association analysis. These methods enable InkSight to automatically identify and describe key patterns and anomalies in the data, not only enhancing the speed of insight generation but also ensuring that generated insights are grounded in robust statistical methods, thereby improving the overall reliability of the process.

Kim et al. [90] explored the use of ML to answer questions about charts and generate visual explanations. They developed an automatic pipeline that extracts data and visual encodings from an input chart, transforms NL questions into queries about the data, and generates visual explanations for the answers. This approach leverages SoTA ML algorithms to handle complex operations such as value retrieval, comparison, and aggregation, significantly reducing the cognitive load on users. The pipeline's ability to generate transparent and useful explanations enhances the interpretability of the insights derived from the charts.

DashBot [30] introduces a deep reinforcement learning model for generating analytical dashboards. It uses RL to explore and imitate human exploration behavior in dashboard creation, leveraging well-established visualization knowledge and the estimation capacity of reinforcement learning. The model is designed to generate dashboards that are both insightful and aesthetically pleasing, with a focus on diversity, parsimony, and the discovery of meaningful data patterns. Through ablation studies and user studies showing that it outperforms existing methods in terms of overall quality, understandability, and insightfulness, the effectiveness of DashBot in insight generation is well demonstrated.

Foresight [39] presents a system that facilitates the rapid discovery of visual insights from large, high-dimensional datasets. It uses a novel framework of insights, insight metrics, and visualizations to guide users through the exploration process. The system employs sketching techniques to achieve interactive performance for insight queries, allowing users to quickly identify and explore strong manifestations of statistical properties in the data. Foresight's approach of focusing on the space of insights rather than the space of data dimensions and visual encodings provides a more efficient and user-friendly experience, enabling analysts to generate and evaluate hypotheses more effectively.

#### *4.3.3. Large-model-based Techniques for Insight Generation*

The rapid advancement of large models has opened new possibilities for insight generation across diverse domains, from structured data analysis to creative content generation. These techniques share a common foundation in leveraging large models' reasoning capabilities while addressing their inherent limitations through innovative representations, interactive frameworks, and automated workflows.

Instead of extracting insights directly from charts, InkSight [59] focuses on facilitating the documentation of chart findings by leveraging user sketches. After users intuitively sketch on visualizations to indicate areas of interest, the system utilizes GPT-3.5 to convert data facts inferred from user sketches into natural language documentation. The integration of LLMs ensures that the generated documentation is coherent and natural, making it easier for users to share and recall their analysis findings.

AutoTitle [106] employs a large-scale natural language transformer model, specifically the T5 model, fine-tuned on a fact-to-title dataset to generate fluent and diverse titles that highlight key insights from visualizations. The system extracts underlying data from visualizations, computes multi-level facts, and uses these facts to generate titles that convey important data features. This approach helps users quickly grasp the main insights from visualizations through automatically generated titles.

VisTR [107] addresses the limitations of directly applying LLMs to time series table reasoning, particularly their lack of trend recognition and analysis capabilities. VisTR transforms time series data facets into visual representations to facilitate pattern recognition and understanding in complex tabular data. By constructing a joint latent space that aligns three modalities-chart, text, and hand-drawn sketches-VisTR enables cross-modal exploration and reasoning. The framework's integration of pruning and indexing mechanisms further ensures scalability, demonstrating the feasibility of using MLLMs and visual representation to help complete an intuitive data insight mining process.

To enhance user experience in extracting insights, LEVA [61] explores the integration of LLMs into VA workflows. It proposes a framework that leverages LLMs to support users in three key stages: onboarding, exploration, and summarization. During onboarding, LLMs interpret visualization designs and relationships to generate tutorials. In the exploration phase, LLMs recommend insights based on the system's status and data, facilitating mixed-initiative exploration. For summarization, LLMs help retrace analytical history and generate insightful reports. The study demonstrates LEVA's effectiveness through usage scenarios and a user study, showing significant improvements in efficiency and accuracy of insight generation compared to traditional methods.

The topic of automated insight generation is fully reflected in InsightLens [80] and InsightPilot [15] . InsightLens takes a conversational perspective, employing a multi-agent architecture to continuously monitor and extract insights from analytic dialogues. Its Insight Extraction Agent identifies meaningful patterns while associating them with supporting evidence, and the Insight Management Agent organizes this knowledge along thematic and data-centric dimensions. This passive, conversation-driven method contrasts with InsightPilot's proactive exploration paradigm, where an LLM-driven engine autonomously designs and executes analysis sequences in response to user queries. Together, these systems showcase the spectrum of possibilities in automated insight generation—from reactive extraction to guided exploration—while significantly lowering the expertise barrier for data analysis.

More recently, ChartInsighter [54] introduces a novel system that leverages LLMs to automatically generate summaries of time-series data charts. It addresses a critical challenge in the field by identifying and mitigating common hallucinations that can occur during the summary generation process. By integrating external data analysis modules and employing multi-agent collaboration, ChartInsighter enhances the accuracy and comprehensiveness of the generated summaries, effectively reducing hallucinations and improving the quality of insights. The system's self-consistency test further ensures the quality of the insights, making ChartInsighter a valuable tool for insight generation.

## 5. Research Challenges and Opportunities

### *5.1. Challenges*

*Lack of High-Quality Datasets.* A critical barrier in chart understanding research is the lack of diverse, large-scale, and accurately annotated datasets that adequately reflect real-world complexity. Most datasets (e.g., DVQA [115], ChartQA [44]) focus on basic chart types (bar, line, pie) but omit complex or hybrid visualizations (e.g., stacked area charts, small multiples, or interactive dashboards). This restricts model generalizability to practical scenarios where charts combine multiple encodings. Also, many chart datasets suffer from annotation inconsistencies that hinder model reliability. For example, axes or legends in charts may be fully annotated, but critical elements (e.g., outliers, trendlines) are ignored, forcing models to "guess" missing context. This leads to overfitting on shallow features (e.g., axis ticks) rather than learning robust semantics.

*Task-specific Limitations.* Even with available datasets, models still struggle with specific tasks. Current annotations often prioritize low-level tasks (e.g., extracting raw values) over higher-order reasoning. For example, datasets rarely label why a trend exists (e.g., causal relationships) or how design choices (e.g., log vs. linear scales) change perception, limiting progress toward interpretative chart understanding. As a result, current approaches often excel in narrow tasks (e.g., extracting numerical values from bar charts) but struggle with broader challenges like interpreting implicit trends, combining multi-chart reasoning, or handling domain-specific visualizations (e.g., biomedical charts), as they require more complex visual reasoning pipelines.

### *5.2. Opportunities*

*Advanced Chart Understanding.* As charts become increasingly sophisticated mediums for data communication, there is a growing interest in developing models capable of decoding complex visual information embedded in hybrid charts and interactive dashboards. These challenges necessitate advancements in visual perception and fine-grained vision-language alignment for MLLMs. Some works have demonstrated that incorporating techniques from multimodal learning (e.g., vision-language pretraining) has the potential to bridge low-level visual features with high-level semantic concepts [122, 107], thus fostering deeper chart understanding. Furthermore, innovative approaches leveraging program-of-thought reasoning have enhanced MLLMs' performance in chart understanding and reasoning scenarios, especially those involving mathematical computations [52, 123]

Future work may systematically discover the chart-specific bottlenecks in current MLLMs and develop targeted solutions to address these limitations. Potential paths include specialized attention mechanisms for chart element recognition and domain-adaptive pertaining strategies to enhance the robustness of models across diverse chart types and domains. Such advancements would continually enhance the visual perception capabilities of MLLMs in chart understanding tasks, and then narrow the gap between models and human performance in chart understanding.

*Applications in Downstream Tasks.* The enhanced fine-grained grounding capabilities of MLLMs in charts can empower various downstream applications, such as automated scientific figure analysis, real-time business report generation, and educational tools for data literacy. In particular, for dynamic interactive charts and dashboards, MLLMs' precise grounding capabilities can directly enable user operations such as 'click', 'filter', or 'selection' to interact with elements of charts. This ability to ground visual elements accurately can enable deeper, more intuitive data exploration.

The visual insight mining pipeline can be extended to various tasks and domains. In finance, it can enhance the analysis of market trends and portfolio performance; for manufacturing and supply chain management, it can help extract abnormal data from dashboards. For different requirements in complex business and scientific scenarios, collaborations with domain experts will be crucial.

*Emerging Research Questions.* The growing focus on chart understanding has sparked plenty of novel research questions that extend beyond traditional tasks like data extraction or basic summarization. A critical question lies in quantifying and mitigating hallucinations that occur when large models generate insights from charts. Addressing this issue calls for the development of specialized evaluation metrics and benchmarks. Another promising path involves the understanding and interpretation of uncertainty in visualizations. Effective reasoning about visual representations of uncertainty (e.g., error bars, confidence intervals) is crucial, particularly when making comparative assertions or trend analyses [124]. Furthermore, the compositional reasoning over chart elements presents a challenge. Many complex insights require compositional understanding across multiple elements. Future work could explore enhancing models' ability to perform multiple-step reasoning that integrates information across these components and improve overall interpretative accuracy.

*Pipeline Innovation.* The advancing capabilities of MLLMs may significantly transform the pipeline of visual insight mining. While conventional pipelines in Figure 1 follow rigid, sequential steps, emerging MLLM-powered approaches enable a simpler pipeline, directly from data to visualization and insight, in an intuitive manner. As shown in Figure 7, future systems may increasingly shift towards user-centric frameworks and incorporate feedback loops, enabling users to interactively query or refine charts via natural lan-

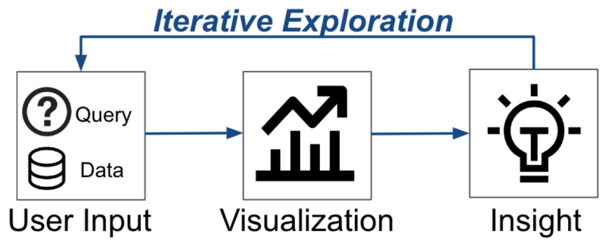


Figure 7: A potential visual insight mining pipeline in MLLM-powered approaches.

guage and mine visual insights iteratively.

## 6. Conclusion

In this survey, we investigate the emerging field of visual insight mining, a domain that has become increasingly vital for transforming complex data into actionable knowledge. The importance of this field stems from visualization's unique ability to make complex data interpretable while modeling the journey from raw data to meaningful insights. Realizing the importance of visualization in the insight mining process, researchers have significantly advanced two main stages: Data2Vis and Vis2Insight. They have also made significant progress in critical tasks such as task decomposition, visualization generation, visualization recommendation, chart parsing, chart question answering, and insight generation. For each of the six tasks, we comprehensively review three primary methods: rule-based approaches, learning-based techniques, and large-model-based solutions. Our survey reveals that visualization serves as a crucial step in the data-to-insight pipeline. More importantly, visualization has great potential to completely bridge the gap between data and insight, especially in this multimodal era.

## References

[1] L. Battle, A. Ottley, What Do We Mean When We Say "Insight"? A Formal Synthesis of Existing Theory, IEEE Trans. Vis. Comput. Graph. 30 (9) (2023) 6075–6088.

[2] K. Wongsuphasawat, Y. Liu, J. Heer, Goals, process, and challenges of exploratory data analysis: An interview study, arXiv preprint arXiv:1911.00568 (2019).

[3] J. W. Tukey, et al., Exploratory data analysis, Vol. 2, Springer, 1977.

[4] W. W. Cooley, P. Lohnes, Multivariate Data Analysis, John Wiley Sons, N.Y., 1971.

[5] J. T. Behrens, Principles and procedures of exploratory data analysis., Psychological Methods 2 (2) (1997) 131–160.

[6] H. Wickham, C. Sievert, ggplot2: elegant graphics for data analysis, Vol. 10, Springer, 2009.

[7] S. Idreos, O. Papaemmanouil, S. Chaudhuri, Overview of data exploration techniques, in: Proc. ACM SIGMOD, 2015, pp. 277–281.

[8] A. Rule, A. Tabard, J. D. Hollan, Exploration and Explanation in Computational Notebooks, in: Proc. ACM CHI, 2018, p. 1–12.

[9] W. S. Cleveland, Visualizing data, Hobart Press, 1993.

[10] E. R. Tufte, P. R. Graves-Morris, The visual display of quantitative information, Vol. 2, Graphics Press Cheshire, CT, 1983.

[11] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer, G. Melançon, Visual analytics: Definition, process, and challenges, in: Information visualization: Human-centered issues and perspectives, 2008, pp. 154–175.

[12] A. Wu, D. Deng, F. Cheng, Y. Wu, S. Liu, H. Qu, In defence of visual analytics systems: Replies to critics, IEEE Trans. Vis. Comput. Graph. 29 (1) (2022) 1026–1036.

[13] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, The kdd process for extracting useful knowledge from volumes of data, CACM 39 (11) (1996) 27–34.

[14] Microsoft, Insight types specification (2016).
URL `https://www.microsoft.com/en-us/research/wp-content/uploads/2016/12/Insight-Types-Specification.pdf`

[15] P. Ma, R. Ding, S. Wang, S. Han, D. Zhang, InsightPilot: An LLM-Empowered Automated Data Exploration System, in: Proc. EMNLP, 2023, pp. 346–352.

[16] J. H. Elder, R. M. Goldberg, Ecological statistics of gestalt laws for the perceptual organization of contours, Journal of Vision 2 (4) (2002) 5–5.

[17] G. A. Agoston, Color theory and its application in art and design, Vol. 19, Springer, 2013.

[18] J. Bertin, Semiology of graphics, University of Wisconsin Press, 1983.

[19] L. Wilkinson, The grammar of graphics, in: Handbook of computational statistics: Concepts and methods, 2011, pp. 375–414.

[20] J. J. Koenderink, A. J. Van Doorn, The singularities of the visual mapping, Biological Cybernetics 24 (1) (1976) 51–59.

[21] H. C. Santiago, Visual mapping to enhance learning and critical thinking skills., Optometric Education 36 (3) (2011).

[22] R. A. Rensink, G. Baldridge, The Perception of Correlation in Scatterplots, Comput. Graph. Forum 29 (3) (2010) 1203–1210.

[23] J. Mackinlay, Automating the design of graphical presentations of relational information, ACM Trans. Graph. 5 (2) (1986) 110–141.

[24] D. Ceneda, T. Gschwandtner, T. May, S. Miksch, H.-J. Schulz, M. Streit, C. Tominski, Characterizing guidance in visual analytics, IEEE Trans. Vis. Comput. Graph. 23 (1) (2016) 111–120.

[25] M. I. Jordan, T. M. Mitchell, Machine learning: Trends, perspectives, and prospects, Science 349 (6245) (2015) 255–260.

[26] X. Wang, Z. Wu, W. Huang, Y. Wei, Z. Huang, M. Xu, W. Chen, VIS+AI: integrating visualization with artificial intelligence for efficient data analysis, Frontiers of Computer Science 17 (6) (2023).

[27] H. Li, Y. Wang, S. Zhang, Y. Song, H. Qu, KG4Vis: A knowledge graph-based approach for visualization recommendation, IEEE Trans. Vis. Comput. Graph. 28 (1) (2021) 195–205.

[28] V. Dibia, c. Demiralp, Data2Vis: Automatic Generation of Data Visualizations Using Sequence-to-Sequence Recurrent Neural Networks, IEEE Comput. Graph. Appl. 39 (5) (2019) 33–46.

[29] K. Hu, M. A. Bakker, S. Li, T. Kraska, C. Hidalgo, VizML: A Machine Learning Approach to Visualization Recommendation, in: Proc. ACM CHI, 2019, p. 1–12.

[30] D. Deng, A. Wu, H. Qu, Y. Wu, Dashbot: Insight-driven dashboard generation based on deep reinforcement learning, IEEE Trans. Vis. Comput. Graph. 29 (1) (2022) 690–700.

[31] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al., GPT-4 technical report, arXiv preprint arXiv:2303.08774 (2023).

[32] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, et al., A Survey on Evaluation of Large Language Models, ACM Transactions on Intelligent Systems and Technology 15 (3) (2024) 1–45.

[33] J. A. Baktash, M. Dawodi, GPT-4: A review on advancements and opportunities in natural language processing, arXiv preprint arXiv:2305.03195 (2023).

[34] P. Maddigan, T. Susnjak, Chat2VIS: Generating Data Visualizations via Natural Language Using ChatGPT, Codex and GPT-3 Large Language Models, IEEE Access 11 (2023) 45181–45193.

[35] V. Dibia, LIDA: A Tool for Automatic Generation of Grammar-Agnostic Visualizations and Infographics using Large Language Models, in: Proc. ACL, 2023, pp. 113–126.

[36] L. Wang, S. Zhang, Y. Wang, E.-P. Lim, Y. Wang, LLM4Vis: Explainable Visualization Recommendation using ChatGPT, in: Proc. EMNLP, 2023, pp. 675–692.

[37] S.-C. Liu, S. Wang, T. Chang, W. Lin, C.-W. Hsiung, Y.-C. Hsieh, Y.-P. Cheng, S.-H. Luo, J. Zhang, JarviX: A LLM No code Platform for Tabular Data Analysis and Optimization, in: Proc. EMNLP, 2023, pp. 622–630.

[38] Y. Tian, W. Cui, D. Deng, X. Yi, Y. Yang, H. Zhang, Y. Wu, Chart-GPT: Leveraging LLMs to Generate Charts From Abstract Natural

Language , IEEE Trans. Vis. Comput. Graph. 31 (03) (2025) 1731–1745.

[39] c. Demiralp, P. J. Haas, S. Parthasarathy, T. Pedapati, Foresight: recommending visual insights, Proc. VLDB Endow. 10 (12) (2017) 1937–1940.

[40] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.

[41] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, Deep learning, Vol. 1, MIT Press Cambridge, 2016.

[42] R. S. Sutton, A. G. Barto, et al., Reinforcement learning: An introduction, Vol. 1, MIT Press Cambridge, 1998.

[43] N. Methani, P. Ganguly, M. M. Khapra, P. Kumar, PlotQA: Reasoning over Scientific Plots, in: Proc. WACV, 2020, pp. 1527–1536.

[44] A. Masry, D. X. Long, J. Q. Tan, S. Joty, E. Hoque, ChartQA: A Benchmark for Question Answering about Charts with Visual and Logical Reasoning, in: Proc. ACL Findings, 2022, pp. 2263–2279.

[45] Z. Yang, L. Li, K. Lin, J. Wang, C.-C. Lin, Z. Liu, L. Wang, The dawn of LMMs: Preliminary explorations with GPT-4V (ision), arXiv preprint arXiv:2309.17421 9 (1) (2023) 1.

[46] J. Wu, W. Gan, Z. Chen, S. Wan, P. S. Yu, Multimodal large language models: A survey, in: Proc. IEEE BigData, 2023, pp. 2247–2256.

[47] X. Yang, Y. Wu, Y. Zhu, N. Tang, Y. Luo, AskChart: Universal Chart Understanding through Textual Enhancement, arXiv preprint arXiv:2412.19146 (2024).

[48] A. Masry, P. Kavehzadeh, X. L. Do, E. Hoque, S. Joty, UniChart: A Universal Vision-language Pretrained Model for Chart Comprehension and Reasoning, in: Proc. EMNLP, 2023, pp. 14662–14684.

[49] Z. Xu, S. Du, Y. Qi, C. Xu, C. Yuan, J. Guo, Chartbench: A benchmark for complex visual reasoning in charts, arXiv preprint arXiv:2312.15915 (2023).

[50] T. Guan, F. Liu, X. Wu, R. Xian, Z. Li, X. Liu, X. Wang, L. Chen, F. Huang, Y. Yacoob, et al., HallusionBench: An Advanced Diagnostic Suite for Entangled Language Hallucination and Visual Illusion in Large Vision-Language Models, in: Proc. ICCV, 2024, pp. 14375–14385.

[51] R. Xia, B. Zhang, H. Ye, X. Yan, Q. Liu, H. Zhou, Z. Chen, P. Ye, M. Dou, B. Shi, et al., ChartX & ChartVLM: A Versatile Benchmark and Foundation Model for Complicated Chart Reasoning, arXiv preprint arXiv:2402.12185 (2024).

[52] L. Zhang, A. Hu, H. Xu, M. Yan, Y. Xu, Q. Jin, J. Zhang, F. Huang, TinyChart: Efficient Chart Understanding with Program-of-Thoughts Learning and Visual Token Merging, in: Proc. EMNLP, 2024, pp. 1882–1898.

[53] S. Kantharaj, R. T. Leong, X. Lin, A. Masry, M. Thakkar, E. Hoque, S. Joty, Chart-to-Text: A Large-Scale Benchmark for Chart Summarization, in: Proc. ACL, 2022, pp. 4005–4023.

[54] F. Wang, B. Wang, X. Shu, Z. Liu, Z. Shao, C. Liu, S. Chen, ChartInsighter: An Approach for Mitigating Hallucination in Time-series Chart Summary Generation with A Benchmark Dataset, arXiv preprint arXiv:2501.09349 (2025).

[55] F. Meng, W. Shao, Q. Lu, P. Gao, K. Zhang, Y. Qiao, P. Luo, ChartAssistant: A Universal Chart Multimodal Language Model via Chart-to-Table Pre-training and Multitask Instruction Tuning, in: Proc. ACL Findings, 2024, pp. 7775–7803.

[56] Y. Han, C. Zhang, X. Chen, X. Yang, Z. Wang, G. Yu, B. Fu, H. Zhang, ChartLlama: A Multimodal LLM for Chart Understanding and Generation, arXiv preprint arXiv:2311.16483 (2023).

[57] J. Finnie-Ansley, P. Denny, B. A. Becker, A. Luxton-Reilly, J. Prather, The robots are coming: Exploring the implications of openai codex on introductory programming, in: Proceedings of the Australasian Computing Education Conference, 2022, pp. 10–19.

[58] Z. Li, H. Miao, V. Pascucci, S. Liu, Visualization literacy of multimodal large language models: A comparative study, arXiv preprint arXiv:2407.10996 (2024).

[59] Y. Lin, H. Li, L. Yang, A. Wu, H. Qu, InkSight: Leveraging Sketch Interaction for Documenting Chart Findings in Computational Notebooks, IEEE Trans. Vis. Comput. Graph. 30 (1) (2023) 944–954.

[60] Q. Wang, Z. Chen, Y. Wang, H. Qu, A survey on ML4VIS: Applying machine learning advances to data visualization, IEEE Trans. Vis. Comput. Graph. 28 (12) (2021) 5134–5153.

[61] Y. Zhao, Y. Zhang, Y. Zhang, X. Zhao, J. Wang, Z. Shao, C. Turkay, S. Chen, LEVA: Using Large Language Models to Enhance Visual Analytics, IEEE Trans. Vis. Comput. Graph. 31 (3) (2025) 1830–1847.

[62] T. Munzner, Visualization analysis and design, CRC Press, 2014.

[63] K. Börner, A. Bueckle, M. Ginda, Data visualization literacy: Definitions, conceptual frameworks, exercises, and assessments, Proceedings of the National Academy of Sciences 116 (6) (2019) 1857–1864.

[64] D. Gotz, D. Borland, Data-driven healthcare: challenges and opportunities for interactive visualization, IEEE Comput. Graph. Appl. 36 (3) (2016) 90–96.

[65] P. Godfrey, J. Gryz, P. Lasek, Interactive visualization of large data sets, IEEE Transactions on Knowledge and Data Engineering 28 (8) (2016) 2142–2157.

[66] M. F. De Oliveira, H. Levkowitz, From visual data exploration to visual data mining: A survey, IEEE Trans. Vis. Comput. Graph. 9 (3) (2003) 378–394.

[67] L. Battle, R. Chang, M. Stonebraker, Dynamic prefetching of data tiles for interactive visualization, in: Proc. ACM SIGMOD, 2016, pp. 1363–1375.

[68] M. T. McCrudden, D. N. Rapp, How visual displays affect cognitive processing, Educational Psychology Review 29 (2017) 623–639.

[69] A. Narechania, A. Srinivasan, J. Stasko, NL4DV: A toolkit for generating analytic specifications for data visualization from natural language queries, IEEE Trans. Vis. Comput. Graph. 27 (2) (2020) 369–379.

[70] T. Wu, M. Terry, C. J. Cai, AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts, in: Proc. ACM CHI, 2022, pp. 1–22.

[71] Y. Zhao, J. Wang, L. Xiang, X. Zhang, Z. Guo, C. Turkay, Y. Zhang, S. Chen, LightVA: Lightweight Visual Analytics with LLM Agent-Based Task Planning and Execution, IEEE Trans. Vis. Comput. Graph. (2024) 1–13.

[72] M. Vartak, S. Rahman, S. Madden, A. Parameswaran, N. Polyzotis, SeeDB: Efficient Data-Driven Visualization Recommendations to Support Visual Analytics, Proc. VLDB Endow. 8 (13) (2015) 2182–2193.

[73] J. Wang, X. Li, C. Li, D. Peng, A. Z. Wang, Y. Gu, X. Lai, H. Zhang, X. Xu, X. Dong, et al., Ava: An automated and ai-driven intelligent visual analytics framework, Visual Informatics 8 (2) (2024) 106–114.

[74] C. Liu, L. Xie, Y. Han, D. Wei, X. Yuan, AutoCaption: An approach to generate natural language description from visualization automatically, in: Proc. IEEE PacificVis, 2020, pp. 191–195.

[75] D. Jung, W. Kim, H. Song, J.-i. Hwang, B. Lee, B. Kim, J. Seo, ChartSense: Interactive data extraction from chart images, in: Proc. ACM CHI, 2017, p. 6706–6717.

[76] M. Savva, N. Kong, A. Chhajta, L. Fei-Fei, M. Agrawala, J. Heer, ReVision: Automated classification, analysis and redesign of chart images, in: Proc. ACM UIST, 2011, pp. 393–402.

[77] Z. Zhou, H. Wang, Z. Zhao, F. Zheng, Y. Wang, W. Chen, Y. Wang, Chartkg: A knowledge-graph-based representation for chart images, IEEE Trans. Vis. Comput. Graph. (2024) 1–15.

[78] E. Hoque, P. Kavehzadeh, A. Masry, Chart question answering: State of the art and future directions, Comput. Graph. Forum 41 (3) (2022) 555–572.

[79] F. Liu, F. Piccinno, S. Krichene, C. Pang, K. Lee, M. Joshi, Y. Altun, N. Collier, J. Eisenschlos, MatCha: Enhancing Visual Language Pre-training with Math Reasoning and Chart Derendering, in: Proc. ACL, 2023, pp. 12756–12770.

[80] L. Weng, X. Wang, J. Lu, Y. Feng, Y. Liu, H. Feng, D. Huang, W. Chen, InsightLens: Augmenting LLM-Powered Data Analysis With Interactive Insight Management and Navigation , IEEE Trans. Vis. Comput. Graph. 31 (06) (2025) 3719–3732.

[81] A. Wu, Y. Wang, X. Shu, D. Moritz, W. Cui, H. Zhang, D. Zhang, H. Qu, AI4VIS: Survey on artificial intelligence approaches for data visualization, IEEE Trans. Vis. Comput. Graph. 28 (12) (2021) 5049–5070.

[82] Y. Ye, J. Hao, Y. Hou, Z. Wang, S. Xiao, Y. Luo, W. Zeng, Generative AI for visualization: State of the art and future directions, Visual Informatics 8 (2) (2024) 43–66.

[83] D. Shi, X. Xu, F. Sun, Y. Shi, N. Cao, Calliope: Automatic visual data story generation from a spreadsheet, IEEE Trans. Vis. Comput. Graph. 27 (2) (2020) 453–463.

[84] W. Epperson, V. Gorantla, D. Moritz, A. Perer, Dead or alive: Continuous data profiling for interactive data science, IEEE Trans. Vis. Comput. Graph. 30 (1) (2023) 197–207.

[85] A. Y. Wang, W. Epperson, R. A. DeLine, S. M. Drucker, Diff in the loop: Supporting data comparison in exploratory data analysis, in: Proc. ACM CHI, 2022, pp. 1–10.

[86] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, J. Heer, Voyager: Exploratory analysis via faceted browsing of visualization recommendations, IEEE Trans. Vis. Comput. Graph. 22 (1) (2015) 649–658.

[87] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, J. Heer, Towards a general-purpose query language for visualization recommendation, in: Proceedings of the Workshop on Human-in-the-loop Data Analytics, 2016, pp. 1–6.

[88] L. Shen, E. Shen, Z. Tai, Y. Song, J. Wang, M. Agus, C. Garth, A. Kerren, Taskvis: Task-oriented visualization recommendation., in: Proc. EuroVis, 2021, pp. 91–95.

[89] J. Poco, J. Heer, Reverse-Engineering Visualizations: Recovering Visual Encodings from Chart Images, Comput. Graph. Forum 36 (3) (2017) 353–363.

[90] D. H. Kim, E. Hoque, M. Agrawala, Answering Questions about Charts and Generating Visual Explanations, in: Proc. ACM CHI, 2020, p. 1–13.

[91] W. Zeng, C. Lin, J. Lin, J. Jiang, J. Xia, C. Turkay, W. Chen, Revisiting the modifiable areal unit problem in deep traffic prediction with visual analytics, IEEE Trans. Vis. Comput. Graph. 27 (2) (2021) 839–848.

[92] A. Vertsel, M. Rumiantsau, Hybrid llm/rule-based approaches to business insights generation from structured data, arXiv preprint arXiv:2404.15604 (2024).

[93] Y. Guo, D. Shi, M. Guo, Y. Wu, N. Cao, Q. Chen, Talk2Data: A natural language interface for exploratory visual analysis via question decomposition, ACM Transactions on Interactive Intelligent Systems 14 (2) (2024) 1–24.

[94] C. Liu, Y. Han, R. Jiang, X. Yuan, ADVISor: Automatic Visualization Answer for Natural-Language Question on Tabular Data, in: Proc. IEEE PacificVis, 2021, pp. 11–20.

[95] Y. Luo, X. Qin, N. Tang, G. Li, DeepEye: Towards Automatic Data Visualization, in: Proc. IEEE ICDE, 2018, pp. 101–112.

[96] Z.-Q. Cheng, Q. Dai, A. G. Hauptmann, Chartreader: A unified framework for chart derendering and comprehension without heuristic rules, in: Proc. ICCV, 2023, pp. 22202–22213.

[97] K. Kafle, R. Shrestha, S. Cohen, B. Price, C. Kanan, Answering questions about data visualizations using efficient bimodal fusion, in: Proc. WACV, 2020, pp. 1498–1507.

[98] J. Hao, M. Yang, Q. Shi, Y. Jiang, G. Zhang, W. Zeng, FinFlier: Automating Graphical Overlays for Financial Visualizations With Knowledge-Grounding Large Language Model, IEEE Trans. Vis. Comput. Graph. (2024) 1–17.

[99] S. Shen, Z. Lin, W. Liu, C. Xin, W. Dai, S. Chen, X. Wen, X. Lan, DashChat: Interactive Authoring of Industrial Dashboard Design Prototypes through Conversation with LLM-Powered Agents, arXiv preprint arXiv:2504.12865 (2025).

[100] G. Li, X. Wang, G. Aodeng, S. Zheng, Y. Zhang, C. Ou, S. Wang, C. H. Liu, Visualization Generation with Large Language Models: An Evaluation, arXiv preprint arXiv:2401.11255 (2024).

[101] H. W. Wang, M. Gordon, L. Battle, J. Heer, DracoGPT: Extracting Visualization Design Preferences from Large Language Models , IEEE Trans. Vis. Comput. Graph. 31 (01) (2025) 710–720.

[102] S. Li, X. Chen, Y. Song, Y. Song, C. J. Zhang, F. Hao, L. Chen, Prompt4Vis: Prompting Large Language Models with Example Mining and Schema Filtering for Tabular Data Visualization, The VLDB Journal 34 (4) (2025).

[103] A. Liew, K. Mueller, Using large language models to generate engaging captions for data visualizations, arXiv preprint arXiv:2212.14047 (2022).

[104] K. Lee, M. Joshi, I. R. Turc, H. Hu, F. Liu, J. M. Eisenschlos, U. Khandelwal, P. Shaw, M.-W. Chang, K. Toutanova, Pix2Struct: Screenshot Parsing as Pretraining for Visual Language Understanding, in: Proc. ICML, 2023, pp. 18893–18912.

[105] X. Zeng, H. Lin, Y. Ye, W. Zeng, Advancing Multimodal Large Language Models in Chart Question Answering with Visualization-Referenced Instruction Tuning, IEEE Trans. Vis. Comput. Graph. 31 (1) (2025) 525–535.

[106] C. Liu, Y. Guo, X. Yuan, AutoTitle: An Interactive Title Generator for Visualizations, IEEE Trans. Vis. Comput. Graph. 30 (8) (2024) 5276–5288.

[107] J. Hao, Z. Liang, C. Li, Y. Luo, j. Li, W. Zeng, VisTR: Visualizations as representations for time-series table reasoning, arXiv preprint arXiv:2406.03753 (2024).

[108] T. Wu, S. He, J. Liu, S. Sun, K. Liu, Q.-L. Han, Y. Tang, A brief overview of ChatGPT: The history, status quo and potential future development, IEEE/CAA Journal of Automatica Sinica 10 (5) (2023) 1122–1136.

[109] R. Mao, G. Chen, X. Zhang, F. Guerin, E. Cambria, GPTEval: A survey on assessments of ChatGPT and GPT-4, arXiv preprint arXiv:2308.12488 (2023).

[110] E. Waisberg, J. Ong, M. Masalkhi, S. A. Kamran, N. Zaman, P. Sarker, A. G. Lee, A. Tavakkoli, GPT-4: a new era of artificial intelligence in medicine, Irish Journal of Medical Science (1971-) 192 (6) (2023) 3197–3200.

[111] D. Moritz, C. Wang, G. L. Nelson, H. Lin, A. M. Smith, B. Howe, J. Heer, Formalizing visualization design knowledge as constraints: Actionable and extensible models in draco, IEEE Trans. Vis. Comput. Graph. 25 (1) (2018) 438–448.

[112] P. Pasupat, P. Liang, Compositional semantic parsing on semi-structured tables, arXiv preprint arXiv:1508.00305 (2015).

[113] Y. Zhang, P. Pasupat, P. Liang, Macro grammars and holistic triggering for efficient semantic parsing, arXiv preprint arXiv:1707.07806 (2017).

[114] S. E. Kahou, V. Michalski, A. Atkinson, Á. Kádár, A. Trischler, Y. Bengio, FigureQA: An Annotated Figure Dataset for Visual Reasoning, arXiv preprint arXiv:1710.07300 (2017).

[115] K. Kafle, B. Price, S. Cohen, C. Kanan, DVQA: Understanding Data Visualizations via Question Answering , in: Proc. CVPR, 2018, pp. 5648–5656.

[116] A. Masry, E. Hoque, Integrating image data extraction and table parsing methods for chart question answering, in: Proc. CVPR ChartQA Workshop, 2021, pp. 1–5.

[117] R. Xia, B. Zhang, H. Peng, H. Ye, X. Yan, P. Ye, B. Shi, Y. Qiao, J. Yan, StructChart: Perception, structuring, reasoning for visual chart understanding, arXiv preprint arXiv:2309.11268 (2023).

[118] F. Liu, X. Wang, W. Yao, J. Chen, K. Song, S. Cho, Y. Yacoob, D. Yu, MMC: Advancing Multimodal Chart Understanding with Large-scale Instruction Tuning, in: Proc. NAACL, 2024, pp. 1287–1310.

[119] L. Hu, D. Wang, Y. Pan, J. Yu, Y. Shao, C. Feng, L. Nie, NovaChart: A Large-scale Dataset towards Chart Understanding and Generation of Multimodal Large Language Models, in: Proc. ACMMM, 2024, pp. 3917–3925.

[120] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al., Chain-of-thought prompting elicits reasoning in large language models, Proc. NeurIPS 35 (2022) 24824–24837.

[121] T. Chen, S. Liu, S. Chang, Y. Cheng, L. Amini, Z. Wang, Adversarial robustness: From self-supervised pre-training to fine-tuning, in: Proc. CVPR, 2020, pp. 699–708.

[122] J. Wei, N. Xu, G. Chang, Y. Luo, B. Yu, R. Guo, mchartqa: A universal benchmark for multimodal chart question answer based on vision-language alignment and reasoning, arXiv preprint arXiv:2404.01548 (2024).

[123] A. Masry, M. Thakkar, A. Bajaj, A. Kartha, E. Hoque, S. Joty, Chart-Gemma: Visual Instruction-tuning for Chart Reasoning in the Wild, in: Proc. COLING, 2025, pp. 625–643.

[124] R. G. Maack, G. Scheuermann, H. Hagen, J. T. H. Peñaloza, C. Gillmann, Uncertainty-aware visual analytics: scope, opportunities, and challenges, The Visual Computer 39 (12) (2023) 6345–6366.