Leveraging Large Language Models for Automated Chart Summarization

Romain Bazin

Department of Electrical & Computer Engineering

McGill University, Montreal

October 1st, 2024

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of

Master of Science

Abstract

Access to graphical information on the internet remains a significant barrier for blind or visually impaired individuals, particularly when it comes to data visualizations like charts. This thesis explores how recent advancements in artificial intelligence can enhance accessibility through innovations in chart summarization, the process of automatically extracting information from a chart and compiling it into a textual summary intelligible to screen readers.

The thesis focuses on two main areas of research. First, it applies the emerging architecture of large language model agents to the task of chart summarization, a novel application in this domain. This approach combines recent advancements in chart information extraction with the reasoning and planning capabilities of large language models. By leveraging natural language processing technologies, it reduces the need for curation of annotated datasets traditionally required to train vision AI models. An implementation of this agent-based approach is developed and evaluated, demonstrating its effectiveness in generating chart summaries.

Second, the thesis addresses the challenge of evaluating chart summarization systems. Adapting approaches from the field of text summarization, it introduces a framework of criteria to assess various aspects of chart summaries quality. This framework serves to identify user preferences while enabling accurate measurement and classification of various chart summarization systems. The framework is assessed through a comparative study with human evaluators, providing insights into its effectiveness for evaluating automated chart summarization systems.

The results of this research will hopefully inform future developments in chart summarization techniques and user-adaptive accessibility solutions.

Abrégé

L'accès aux informations visuelles sur internet demeure un obstacle majeur pour les personnes aveugles ou malvoyantes, en particulier concernant les visualisations de données telles que les graphiques. Cette thèse explore comment les récents progrès en intelligence artificielle peuvent améliorer l'accessibilité grâce à des innovations dans la synthèse de graphiques, un processus d'extraction automatique d'informations à partir d'un graphique et de compilation en un résumé textuel intelligible pour les lecteurs d'écran.

La thèse se concentre sur deux principaux axes de recherche. Premièrement, elle applique l'architecture émergente des agents basés sur les grands modèles de langage à la tâche de synthèse de graphiques, une application novatrice dans ce domaine. Cette approche combine les avancées récentes en extraction d'informations graphiques avec les capacités de raisonnement et de planification des grands modèles de langage. En s'appuyant sur les technologies de traitement du langage naturel, elle réduit le besoin de curation des jeux de données annotés traditionnellement nécessaires pour entraîner les modèles de vision par intelligence artificielle. Une implémentation de cette approche basée

sur des agents est développée et évaluée, démontrant son efficacité dans la génération de résumés de graphiques.

Deuxièmement, la thèse aborde le défi de l'évaluation des systèmes de synthèse de graphiques. En adaptant des approches du domaine de la synthèse de texte, elle introduit un des critères pour évaluer divers aspects de la qualité des résumés de graphiques. Ces critères permettent d'identifier les préférences des utilisateurs tout en permettant une mesure et une classification précises des différents systèmes de synthèse de graphiques. Cette méthode est appliquée dans une étude comparative avec des évaluateurs humains, révélant son utilité pour comparer les modèles et identifier les préférences entre différents groupes d'utilisateurs.

Les résultats de cette recherche pourront, nous l'espérons, éclairer les futurs développements dans les techniques de synthèse de graphiques et les solutions d'accessibilité adaptatives pour les utilisateurs.

Acknowledgements

I extend my deepest gratitude to my supervisor, Professor Jeremy Cooperstock, for his invaluable mentorship, support, and provision of essential infrastructure throughout this journey. His encouragement to follow my research interests has been fundamental to my work.

My heartfelt thanks go to my colleagues at the Shared Reality Lab for their insightful suggestions and advice. Special appreciation is due to Cyan Kuyo, whose consistent guidance has been instrumental to my progress.

I am profoundly grateful to my family for their unwavering support. My parents fostered a nurturing environment that paved the way for my achievements, while my sister's affectionate nature has consistently provided comfort and strength. I extend my heartfelt thanks to Cyril, my best friend, whose faithful friendship has been a constant source of encouragement throughout the years. Finally, I reserve my deepest appreciation for Jade, whose enduring love and support have been essential to my journey and success.

This journey would not have been possible without the collective support and

encouragement of each one mentioned, and for that, I am eternally grateful.

I would also like to acknowledge the helpful assistance of large language models in refining this thesis. In particular, Claude 3.5 Sonnet has been valuable in helping me rephrase and improve my writing, contributing to the clarity and coherence of this work. While the ideas and research are entirely my own, this tool has enhanced the presentation of my thoughts and findings.

Contents

List of Figures

List of Tables

List of Acronyms

AI	Artificial Intelligence.
BVIP	Blind or Visually Impaired People.
COT	Chain of Thought.
CVQA	Chart Visual Question Answering.
GPU	Graphics Processing Unit.
ICL	In-Context Learning.
LLM	Large Language Model.
QC	Quality Criteria.
TOT	Tree of Thoughts.
VFM	Visual Foundation Model.
VIT	Vision Transformer.
VLLM	Vision Large Language Model.

Contributions

This thesis applies LLM-based agent architecture to chart summarization, addressing dataset scarcity in this domain. The approach combines chart information extraction techniques with large language model reasoning capabilities, offering a modular solution for generating summaries without relying on annotated datasets. While not outperforming the closed-source GPT-4-Vision, our method demonstrates competitive results, particularly in linguistic aspects, and surpasses the previous open-source state-of-the-art, Unichart. This contribution provides academia and open-source developers a pathway to potentially match proprietary model performance through improved information extraction and advanced open-source LLMs like LLAMA-3.

Our research also identifies differences in chart summary preferences between expert and novice users. Experts value a balance of linguistic and data-centric aspects, while novices prioritize relevance. These findings contribute to understanding diverse user needs in chart summarization and indicate potential for developing adaptive systems tailored to varying levels of expertise.

Chapter 1

Introduction

I apologize for the oversight. You're right, and I should have maintained the citations. Here's the revised version with the citations included:

Artificial intelligence (AI) is significantly impacting industries and creating new possibilities across diverse domains. From healthcare and education to transportation and beyond, AI is driving substantial innovation and efficiency improvements. Two subfields at the forefront of this development are computer vision and natural language processing. Computer vision focuses on enabling machines to interpret and understand visual information, while natural language processing aims to teach computers to comprehend and communicate in human language.

The advancements in computer vision and natural language processing have created new opportunities for AI-driven accessibility solutions. An important challenge at the intersection

of these fields is the automatic summarization of visual data representations, particularly charts and graphs. While graph summarization has seen significant progress, largely due to advancements in image captioning techniques and the availability of large-scale datasets such as Common Objects in Context (COCO) and Flickr30k, the summarization of charts presents unique challenges that warrant further research [1, 2].

Charts are a prevalent means of conveying complex data and relationships in a visual format, used extensively in scientific publications, business reports, and online media. Unlike graphs, which primarily represent relationships between entities, charts often combine visual elements, textual information, and numerical data values. This multimodal nature makes chart summarization challenging, requiring advanced AI techniques to interpret and synthesize information accurately. This challenge is notable in the context of accessibility for blind and visually impaired (BVI) individuals. While the text surrounding a chart may provide some context, it often doesn't fully capture the information presented in the visual representation. Despite recent progress in chart summarization techniques, many current solutions still struggle to fully convey the depth of information in a format accessible to BVI users [3, 4]. This gap in accessibility leaves BVI users potentially excluded from accessing important information, limiting their ability to fully engage with data-rich content.

Generating comprehensive and accurate summaries of charts is a challenging AI task, requiring an understanding of diverse chart types, data encodings, and represented topics. A

key bottleneck in developing effective chart summarization systems is the need for large, high-quality training datasets. The intricacy and variety of charts demand substantial amounts of labeled examples to train models that can generalize well to the vast space of unseen charts. However, collecting such datasets through manual human annotation is a time-consuming and expensive process. Despite recent progress in chart summarization techniques, many current solutions still struggle to fully convey the depth of information in a format accessible to BVI users [3,4]. This presents a significant challenge for progress in this area, as the cost and effort required to build suitable training data are often considerable.

The first key objective of this thesis is to develop a novel paradigm for chart summarization that harnesses the power of large language models (LLMs). By leveraging the extensive general knowledge and reasoning capabilities of LLMs, this approach aims to reduce the reliance on large labeled training datasets, thus overcoming a critical limitation of prior work.

While prior research has made strides in extracting information from charts, less attention has been devoted to the equally important task of synthesizing this information into coherent, insightful summaries that align with user preferences. The lack of well-defined, measurable attributes for assessing summary quality has hindered progress in understanding what makes a chart summary truly effective and valuable to users. To address this gap, the second key objective of this thesis is to establish a framework of quantitative criteria for evaluating the quality of chart summaries. Inspired by recent advancements in text summarization

evaluation, we propose a set of criteria, each rated on a 1-5 scale, that capture essential aspects of summary quality.

To address these objectives and present our research, this thesis is organized into five chapters

Chapter 2: Background reviews the current state of chart understanding. It examines recent developments in computer vision and natural language processing techniques, while also discussing the limitations of existing chart summarization approaches, with particular attention to the scarcity of high-quality datasets.

Chapter 3: Design of LLM Agent for Chart Summarization introduces the primary contribution of this thesis: an LLM-based agent architecture for chart summarization. It provides a detailed description of the brain, perception, and action modules that constitute this system.

Chapter 4: System Evaluation assesses the performance of the proposed approach against relevant baselines. It introduces a quality criteria framework adapted from text summarization and presents a user study demonstrating its efficacy in capturing nuanced differences in summary quality across models and user groups.

Chapter 5: Discussion analyzes the key findings on LLM-based chart summarization. It considers the limitations of the proposed approach and identifies potential future research directions. The chapter concludes by revisiting the initial objectives and exploring their implications for improving access to visual data.

Chapter 2

Background

2.1 Advances in Computer Vision and Natural Language Processing for Chart Understanding

2.1.1 The Importance of Chart Understanding in Improving Accessibility for Blind or Visually Impaired People

Chart understanding is a subfield of image understanding focused on automatically extracting and interpreting information from charts to facilitate user comprehension. It has a long history of research interest due to its importance in making complex data more understandable. Early work included knowledge-based systems like WERP in the 1980s, which generated weather reports from charts, and efforts in the 2000s to associate text and

graphics in scientific charts for more comprehensive semantic understanding [5, 6].

Chart understanding is important for increasing accessibility for blind or visually impaired people (BVIP), who face significant challenges when accessing visual information in charts. While BVIP rely on assistive technologies like screen readers to navigate digital content, these tools often struggle to convey the full depth of insights and nuances contained within complex visual representations. The surrounding text may provide some context, but it rarely captures all the information encoded in the visual elements, such as trends, patterns, data points relationships, and intended message. Consequently, BVIP are frequently excluded from accessing critical information, hindering their ability to fully engage with the content and participate in data-driven discussions [7].

To address this accessibility gap, researchers have explored various approaches to extracting meaningful information from charts and presenting it in alternative formats that are more accessible to BVIP. These efforts include developing techniques for automatically parsing and analyzing chart images to identify key components such as axes, labels, and data points; a task commonly known as chart understanding [3, 8–11]. By extracting this structured information, it becomes possible to generate alternative renderings of charts that convey the same insights through non-visual modalities, such as audio signals, vibrotactile feedback or force feedback haptics [12, 13]. The ultimate goal is to create a seamless and engaging experience for BVIP by combining these alternative renderings into a coherent and informative format [14–16].

Image-to-text chart understanding is a promising approach to bridging the accessibility gap for BVIP by translating chart images into textual representations. It encompasses three main tasks: chart visual question answering (CVQA), chart derendering, and chart summarization. CVQA involves providing accurate answers to natural language questions about a chart, requiring an understanding of both visual elements and textual content. Chart derendering, also known as chart-to-table, converts a chart image into a structured data table by identifying and extracting components such as bars, lines, legends, and axes. Chart summarization, the primary focus of this thesis, generates concise textual summaries that capture the key insights and salient information in a chart by analyzing visual and textual elements to produce a narrative highlighting the most important aspects of the data.

Among these tasks, chart summarization has received less attention in the research community but provides significant value for BVIP. By generating informative, well-articulated text summaries encapsulating the main takeaways of a chart, chart summarization enables BVIP to access and comprehend the essential information conveyed in visual representations. This thesis aims to advance chart summarization methods by developing new methods in summary generation and evaluation.

The methods employed in image-to-text chart understanding, particularly in chart summarization, draw inspiration from advancements in the field of image understanding [17]. The next section will present the evolution of image captioning and chart summarization methods, exploring the architectures and techniques that have been

adapted from image understanding to tackle the challenges of extracting and conveying information from charts in a textual format.

2.1.2 Evolution of Image Understanding and Chart Understanding Methods

A comprehensive survey on image captioning methods, a task closely related to chart summarization, provides theoretical insights into the architectures used in image understanding systems. The image captioning pipeline typically consists of two main components: a *visual detector* for extracting factual information from visual elements, and a *description generator* for producing descriptions, answers, or tables based on the extracted visual information [18].

Over the years, the implementation of these components has evolved significantly. Early machine learning-based approaches included template-based methods, which relied on predefined templates and statistical language models to generate captions using object detectors [19]. This approach was applied to chart summarization, using a combination of neural networks for visual detection and three-part templates for description generation, consisting of a premier (describing basic shape and factual information), a core (conveying the intended meaning of the chart), and a wrapper (providing additional details for certain chart types) [10]. However, these methods were limited by their reliance on manually defined features and templates, restricting their generalization capabilities. Retrieval-based

methods focused on ranking existing captions or images based on their similarity to the input, selecting the most relevant ones from a predefined dataset [20]. The limitations here were the dependency on dataset diversity and size for caption relevance and novelty, and the reliance on large labeled datasets with simple, short captions, making them unsuitable for chart summarization.

In recent years, there has been a paradigm shift towards deep learning-based approaches for image captioning. Encoder-decoder models, inspired by machine translation systems, encode visual information using convolutional neural networks (CNNs) and decode it into textual descriptions with recurrent neural networks (RNNs) [21–23]. This end-to-end approach allows for more flexible and contextually relevant captions by learning features directly from data. It has been applied to chart summarization in works like STLCQA [24]. Attention-based models further refine the encoder-decoder architecture by enabling the model to focus on specific parts of the image while generating each word of the caption, mimicking human visual attention [25]. This approach leads to more detailed and accurate descriptions and has been extensively used for CVQA and chart derendering models, such as ChartQA, Matcha, and Deplot models, but also in chart summarization as in the Chart-to-Text and Unichart models [3, 4, 11, 26, 27].

2.1.3 Dataset Scarcity in Chart Summarization

While Deep Learning methods have become the dominant approach in image captioning, CVQA, and chart derendering, they require large amounts of high-quality labeled data to reach their full potential. Unfortunately, as highlighted in a review on chart classification and captioning, the main limitation in applying these techniques to chart summarization is the scarcity of such datasets [17]. Recent efforts like Chart-to-Text have aimed to create a large labeled dataset with over 35k chart summaries [4]. However, the dataset suffers from issues with the label summaries containing information not present in the charts, leading to models trained on it exhibiting severe hallucination problems [28]. These models generate coherent but false information about new charts, undermining their real-world usability.

To address dataset scarcity, the creators of the Unichart model proposed crafting a synthetic dataset from a harmonized corpus of real-world charts [3]. They collected 627,000 charts from various sources, including online databases, publications, and existing datasets. For charts with available data tables, they utilized existing datasets or extracted data from SVG elements. They also performed data augmentation by creating new charts from public data tables using visualization libraries. For charts without data tables, they employed object detection and optical character recognition (OCR) to extract textual and visual elements. To generate summaries for the charts, they used a knowledge distillation approach. First, they used a large language model, InstructGPT, to generate summaries for a subset of charts based on extracted textual information, which were then reviewed

and validated by human annotators [29]. These summaries were used to fine-tune another model, Flan-T5 XL, which subsequently generated summaries for the larger dataset [30]. The model trained on the larger dataset, using an attention-based encoder-decoder architecture, achieved state-of-the-art results in chart summarization. However, this large-scale dataset was not made public, so the problem of dataset scarcity persists.

2.1.4 Opportunities from Chart Visual Question Answering

In contrast to the scarcity of chart summarization datasets, the fields of CVQA and chart derendering have thrived due to the emergence of multiple high-quality datasets that are easier to create. Notable CVQA datasets include DVQA, FigureQA, LeafQA, PlotQA, ChartQA, and OpenCQA [26, 31–35]. The availability of these datasets has led to a proliferation of research on CVQA and chart derendering, ranging from early systems combining manual and automated chart specification for exploration and question answering to more recent work like MatCha and DePlot [11, 27, 36].

MatCha, enhances visual language pretraining with math reasoning and chart derendering tasks. MatCha is built upon an image-to-text transformer architecture and learns to extract information from charts by predicting the underlying data table or the code used to render the chart. Additionally, it is pretrained on math reasoning tasks using datasets like MATH and DROP, enabling it to perform numerical reasoning on the extracted information. This pretraining allows MatCha to achieve state-of-the-art results

on several CVQA benchmarks.

DePlot extends the chart derendering capability of MatCha by finetuning the model on a larger dataset of chart-table pairs, achieving even greater accuracy in extracting structured data from charts. Furthermore, DePlot improves upon MatCha visual language reasoning capabilities by proposing a modular approach that separates the task into two steps: (1) chart-to-table conversion using the DePlot model, and (2) reasoning over the extracted table using a LLM. In the first step, the DePlot model converts the chart image into a structured table format. The output of this model is then fed into an LLM, which is prompted to answer questions or perform reasoning tasks based on the extracted table. By leveraging the few-shot learning capabilities of LLMs, DePlot achieves considerable results on CVQA tasks with minimal human-labeled examples, highlighting the promise of this modular approach in scenarios where labeled data is scarce [27].

Following these works, the Unichart model utilized MatCha pretraining scheme and DePlot's finetuning method on their private large-scale dataset to reach even higher accuracy on CVQA and chart derendering tasks.

This presents an opportunity to leverage the advances and resources in CVQA to improve chart summarization. By integrating highly accurate CVQA and derendering models into an automated summarization workflow, we can potentially craft more accurate, coherent, and comprehensive summaries, even in the face of limited labeled summarization datasets.

2.1.5 Conclusion

In conclusion, while the field of chart understanding has made substantial progress by adopting deep learning methods from general image captioning, chart summarization still faces significant challenges. The scarcity of large, high-quality labeled datasets for chart summarization hinders the training of advanced models, in contrast to the related subfields of CVQA and chart derendering, which have benefited from the availability of numerous datasets. This disparity presents an opportunity to leverage models and datasets from CVQA and derendering to improve chart summarization, potentially reducing the reliance on expensive labeled summarization data.

However, utilizing CVQA and chart derendering models for chart summarization introduces two key challenges. Firstly, chart derendering models generate textual outputs that must be translated into coherent and fluent textual summaries, requiring an interface to perform this translation. Due to the lack of labeled chart summary data, deep learning and retrieval-based methods are not viable options. While template-based methods have been used in previous work, they lack generalizability. Secondly, CVQA models require textual input in the form of questions, which depend on the specific chart and are difficult to automate.

The recent emergence of powerful LLMs, made widely accessible through private APIs, presents a potential solution to these challenges. LLMs possess strong natural language capabilities that could enable them to effectively translate the outputs of CVQA and chart

derendering models into coherent textual summaries. Additionally, LLMs could dynamically generate appropriate input questions for CVQA models based on the chart at hand.

To leverage LLMs for automated, robust, and generalizable chart summarization, we propose the development of an LLM-based AI agent. In this context, an AI agent refers to a system that uses LLMs to intelligently process and integrate information from multiple sources (such as CVQA and derendering models) to generate comprehensive chart summaries. The following section will introduce the concept and architecture of this LLM-based AI agent in more detail.

2.2 Large Language Model Agents and Machine Learning Models as Tools

2.2.1 Concept and History of AI Agents

AI agents are artificial entities that perceive their environment, make decisions, and take actions to achieve specified goals. A key characteristic of AI agents is their autonomy in sensing, reasoning, and acting. The concept of AI agents has evolved significantly over the decades, with different approaches emerging to enable increasingly sophisticated agent capabilities [37].

The early era of AI agents in the 1970s-80s was dominated by symbolic AI approaches [38,39]. These agents utilized logical rules and symbolic knowledge representations to reason

and draw inferences. While highly interpretable, symbolic AI agents struggled to handle the uncertainty and complexity of real-world environments, and faced challenges in scalability.

In response to the limitations of symbolic AI, reactive agents were developed in the 1980s-90s [40, 41]. Reactive agents emphasized producing quick, real-time responses based on direct mappings between perceptual inputs and action outputs. Rather than maintaining complex internal models, reactive agents relied on close coupling with the environment. While effective for some tasks, reactive agents had limited capacity for reasoning and long-term planning.

As computational power increased at the turn of the century, reinforcement learning (RL) emerged as a powerful framework for creating AI agents that learn through interaction with an environment [42]. RL agents aim to learn policies that maximize the cumulative reward obtained over time. The integration of deep learning with RL enabled agents to learn highly sophisticated policies [43]. However, challenges remain in the sample efficiency and stability of deep RL agents when applied to complex real-world environments.

Recent advancements in transfer learning and meta-learning have enhanced the learning capabilities of AI agents to some extent. Transfer learning enables agents to leverage knowledge from previous tasks to more efficiently learn new tasks, while meta-learning takes this a step further by learning the learning algorithms themselves to enable quick adaptation to new tasks. However, these approaches still face limitations. Transfer learning can be ineffective when there are significant disparities between the source and target

tasks, potentially leading to negative transfer. Meta-learning also requires substantial pre-training and large sample sizes, making it challenging to learn a truly general policy that can handle any task [44, 45].

The advent of large language models (LLMs) in late 2022 and early 2023 has opened up exciting new possibilities for creating highly capable AI agents [29, 46]. LLMs have demonstrated impressive emergent abilities, such as advanced language understanding, reasoning, and generation, making them well-suited to serve as the foundation for AI agents. Techniques like chain-of-thought reasoning, which involves generating intermediate reasoning steps, and problem decomposition, which breaks down complex tasks into smaller sub-tasks, allow LLM-based agents to exhibit symbolic reasoning and planning skills on par with classical approaches [47–50]. Moreover, few-shot and zero-shot learning enable these agents to efficiently generalize to new tasks with minimal additional training. As a result of these enhanced capabilities, LLM-based agents have achieved remarkable performance on complex real-world applications. For example, agents like ChatDev can assist with software development tasks such as design, coding, testing, and documentation, operating either autonomously or in collaboration with human developers [51].

To fully realize the potential of LLM-based agents for this task, it is key to carefully design an architecture that integrates the necessary components and capabilities. In the following section, we will examine the core elements of an LLM-based AI agent system, detailing the key modules responsible for knowledge representation, reasoning, planning, and interaction.

By delineating the roles and interplay of these components, we aim to establish a solid foundation for the development of an effective chart summarization agent.

2.2.2 Large Language Models as AI Agents

Recent advancements in LLMs have paved the way for the development of capable AI agents. These LLM-based agents leverage a modular architecture, consisting of a central brain module that interacts with perception and action modules, to understand and interact with their environment effectively [37].

2.2.2.1 Brain Module of LLM-based AI Agents

The brain module, centered around an LLM, serves as the hub for various cognitive capabilities, including natural language interaction, knowledge representation, memory management, reasoning, planning, and generalization [37]. LLMs excel at engaging in multi-turn conversations, generating coherent text, and comprehending the intentions behind language to a certain extent [46]. Instruction tuning and few-shot learning enable LLM-based agents to generalize to new tasks without additional training, a capability known as zero-shot generalization. This flexibility is particularly valuable for tasks like chart summarization, which may not have been included in the pretraining or finetuning data [48, 52].

To enhance the reasoning and planning capabilities of LLM-based agents, techniques

such as chain-of-thought prompting, self-consistency methods, and task decomposition have been developed. Chain-of-thought prompting encourages the model to explicitly establish a thought process before generating answers, while self-consistency methods enable the model to explore multiple reasoning paths for optimal decision making [47,49]. Task decomposition breaks down complex problems into more manageable sub-tasks, making problem-solving more efficient and effective.

However, challenges remain in understanding implicit information and mitigating hallucinations, where the model generates inconsistent or factually incorrect content [53, 54]. Memory management is also a crucial aspect of the brain module, as the amount of memory an LLM can process at once is limited by the size of its context window. Methods such as text truncation, input segmentation, attention mechanism modifications, and memory summarization techniques have been proposed to address this limitation [55, 56].

2.2.2.2 Perception Module of LLM-based AI Agents

The perception module is a critical component of LLM-based AI agents, enabling them to receive and process information from various sources and modalities. This perceptual space allows the agents to understand their environment, and make informed decisions [37].

Textual input is the native modality that LLMs are highly proficient at processing. LLMs have a strong inherent ability to communicate with humans through text and can

effectively understand explicit instructions. They also have some capability to perceive implied meanings and intentions behind the text, although this remains an ongoing challenge [46].

While LLMs inherently lack the ability to directly process visual information like images, integrating visual perception significantly expands the agent's understanding of the world. One approach to enable LLMs to understand visual information is to combine an image encoder, such as those based on the vision transformer (ViT) architecture, directly with the LLM to perceive visual content [57]. The image is divided into patches, linearly projected, and treated as input tokens for the transformer. By calculating self-attention between tokens, information from the entire image can be integrated. While this end-to-end training of the visual encoder and LLM can achieve remarkable visual perception abilities, it comes at a substantial computational cost. Large language models that have been extended with these visual perception capabilities are referred to as vision large language models (VLLMs). A prominent example of a VLLM is GPT-4-Vision from OpenAI, which incorporates an image encoder to gain the ability to natively process and understand visual information [58].

A more efficient paradigm is to use extensively pre-trained visual encoders and LLMs, freezing one or both of them during training to balance computational resources and model performance. However, an extra learnable interface layer is typically required to align the visual encoder's output with the LLM's input embeddings. For example, the querying transformer (Q-Former) module has been used as an intermediate layer to extract

language-informative visual representations [59].

Despite these advancements, current methods for visual-linguistic learning in LLMs tend to overfit to the pretraining dataset and struggle to generalize or few-shot learn in new domains. This poses a particular challenge for niche applications like chart understanding, where large-scale datasets for pretraining a visual encoder are not readily available. The lack of a sufficiently diverse and comprehensive dataset of charts hinders the development of a robust visual perception module for this specific domain [17, 37].

Given these limitations, an alternative approach to expanding the perception space of LLMs to the visual domain is by utilizing external tools with computer vision capabilities. This is made possible by the action module of the LLM-based agent, which enables the agent to interface with specialized visual processing tools to virtually extend its perception capabilities without the need for large-scale pretraining datasets or computationally intensive end-to-end training.

2.2.2.3 Action Module of LLM-based AI Agents

While LLMs have demonstrated remarkable abilities in natural language understanding and generation, they often lack the domain expertise or computational capabilities to directly complete complex real-world tasks. One promising approach to address these limitations is enabling LLM-based agents to actively interact with their environment using external tools [37].

The ReAct framework, proposed by Yao et al. (2023), is a methodology that enables LLMs to reason about and utilize specialized tools. It achieves this by combining reasoning and acting. ReAct augments the action space of an LLM agent to include both natural language generation and tool-specific actions. By interleaving reasoning traces and tool interactions, the agent can dynamically compose information, track progress, and adapt its plans based on the evolving context. The thoughts guide the selection and application of tools, while the actions interface with the external tools to gather information or effect changes in the environment. Through this iterative process, the agent can break down complex tasks, reason about intermediate results, and construct robust and interpretable task-solving trajectories [60, 61]. Due to its simplicity and impressive performance, the ReAct framework has been widely adopted in subsequent works [56, 62, 63].

Building upon the ReAct framework, Qin et al. (2023) introduced ToolLLM, a framework for reinforcing open-source LLMs' capabilities in tool use through fine-tuning on a large-scale tool-augmented dataset [64]. Central to the ToolLLM framework is the construction of ToolBench, an instruction-tuning dataset covering over 16,000 APIs. ToolBench characterizes each tool to the LLM by providing detailed documentation in a structured JSON format, including functionality descriptions, required parameters, and example responses. By pre-training on this rich tool dataset, LLMs can generalize their tool use capabilities to new, unseen tools through in-context learning, where the new tool is characterized in the same JSON format used during training. In addition to the fine-tuning

framework, ToolLLM also proposes an alternative to the ReAct reasoning approach called depth-first search based decision tree (DFSDT) reasoning. In DFSDT, the agent explores multiple reasoning paths, retracts unproductive steps, and strategically navigates to the most promising path. While drastically more computationally intensive, this approach enhances the planning and reasoning abilities of the LLM. While the ToolLLM framework has helped open-source models reduce the gap with closed-source models in terms of tool use capabilities, the latter remain significantly superior. In particular, GPT-4 currently achieves state-of-the-art results on the ToolBench benchmark, far outperforming open-source models like ToolLLaMA.

Recent work on Prismer showcases how tool use can enable LLM-based agents to achieve state-of-the-art performance on vision-language reasoning tasks with orders of magnitude less data than prior approaches. By leveraging an ensemble of pre-trained vision, language, and multi-modal expert models as tools, Prismer efficiently pools their specialized knowledge to bootstrap strong multi-modal reasoning capabilities. The expert resampler and adaptor modules proposed in Prismer exemplify effective techniques for integrating multiple expert tools while preserving their pre-trained knowledge. Prismer's strong results demonstrate that tool use, when combined with judicious architecture design, can enable highly sample-efficient learning, robustness to noisy tools, and impressive zero-shot and few-shot generalization. These findings underscore the promise of tool use as a paradigm for modularizing multi-modal learning and facilitating knowledge transfer from pre-training to downstream tasks [65].

Importantly, the action module provides a means to extend an LLM-based agent's perception capabilities beyond its native language modality. By interacting with visual processing tools, the agent can indirectly perceive and reason about images without requiring a specialized visual encoder to be integrated into its core architecture, offering a pragmatic alternative to the challenging and data-intensive process of directly training an LLM to encode images.

A prime example of such an LLM-based agent with extended visual perception through tool use is Visual ChatGPT [66]. This system integrates a diverse set of visual foundation models (VFMs) spanning vision and vision-language tasks with ChatGPT using a prompt manager module. The VFMs cover a wide range of visual processing capabilities, including visual question answering, image captioning, object detection, image generation, and image editing. The prompt manager, equivalent to the brain module in the framework from Xi et al. (2023), serves as a bridge between the language and vision modalities, converting visual information into language instructions for selecting, executing, and chaining the VFMs based on the dialogue context [37]. This enables Visual ChatGPT to perform tasks such as answering questions about image content, generating images from textual descriptions, and executing complex image editing operations.

Through multi-turn discussion experiments and case studies, Visual ChatGPT demonstrated impressive multimodal conversational and visual reasoning capabilities. It was able to handle multi-step visual editing instructions by sequencing relevant VFMs. For

example, given a user request to "detect the depth map of the image first, show the depth image, then generate a new image based on this depth map, and finally describe the final image", Visual ChatGPT first used an depth VFM to compute the gradient-based depth of the original image, an image-to-image VFM to generate from the depth image a new scene with the same character and objects, and finally a captioning VFM to describe the new scene of the image. In other experiments, the system also showed proficiency at answering questions about image content and style, such as identifying objects, describing their relationships, and making precise modifications to the image based on language input. The authors noted that the modular tool-based approach offers flexibility and extensibility advantages, allowing new VFMs to be easily incorporated to expand the agent's capabilities.

2.2.3 Conclusion

In conclusion, large language models have emerged as a powerful foundation for creating highly capable AI agents. By leveraging an architecture consisting of a central brain module that interacts with perception and action modules, these LLM-based agents can understand and interact with their environment to complete complex tasks in full autonomy.

The brain module, powered by an LLM, enables language understanding, knowledge representation, reasoning, and generative capabilities. Techniques like chain-of-thought prompting, self-consistency, and task decomposition can further enhance the reasoning and

planning abilities of these agents. The perception module extends the agent's understanding to the visual domain, either through direct integration of visual encoders or interaction with external computer vision tools. Finally, the action module empowers agents to actively engage with their environment using specialized tools, significantly expanding their task-solving capabilities.

This modular tool-based paradigm offers compelling advantages for complex applications like chart summarization for the blind and visually impaired. By interfacing with domain-specific machine learning models for information extraction from charts, an LLM-based agent can gain the ability to perceive and reason about visual data without requiring a specialized visual encoder or large-scale pretraining. The agent's strong language skills can then be leveraged to synthesize human-like text summaries capturing the key insights from the chart.

The impressive results achieved by recent systems like Visual ChatGPT and Prismer underscore the promise of this approach. By thoughtfully designing the prompt management and tool integration, it is possible to create agents that can engage in substantive discussions about images and even execute complex multi-step visual reasoning and editing tasks. As research in this area progresses, we can expect LLM-based agents to become increasingly adept at understanding and interacting with the real world, opening up exciting opportunities for AI-driven accessibility solutions.

2.3 Limits of Chart Summarization Evaluation Methods

2.3.1 Current Evaluation Methods in Chart Summarization and Their Limitations

Evaluation is central in validating research hypotheses and driving progress in any field. In chart summarization, evaluation aims to measure the quality of generated summaries, but faces significant challenges in deriving reliable, well-aligned metrics from this objective. This section examines current evaluation approaches in chart summarization, their limitations, and opportunities for improvement.

At its core, an evaluation procedure optimizes an objective - in this case, generating high-quality chart summaries. Abstractive text summarization aims to compress long textual documents into a short, human-readable form that contains the most important information from the source by detecting salient parts and paraphrasing them to form the final output, a concept that extends well to chart summarization [67]. However, measuring this objective poses difficulties. Effective metrics should be specific, aligning closely with the objective, and reliable, producing consistent results under identical conditions.

Evaluation methods for chart summarization fall into two main categories: quantitative heuristic scores and human evaluator pairwise comparisons. Heuristic metrics, such as

Bilingual Evaluation Understudy (BLEU), Recall-Oriented Understudy for Gisting Evaluation (ROUGE), and Consensus-based Image Description Evaluation (CIDEr), assess a candidate summary against a reference summary by comparing words or word groups [4]. While highly interpretable, these metrics heavily penalize paraphrasing and alternative word choices, limiting their flexibility. BERTScore offers an alternative by comparing vector embeddings from a pre-trained BERT model, aiming to capture semantic similarity over exact wording [68]. However, the opacity of neural networks renders this method less interpretable. All these metrics also require labeled reference summaries, which are scarce in chart summarization datasets. Consequently, heuristic scores often align poorly with the evaluation objective and show weak correlation with human judgments of summary quality [3, 69].

Human evaluator pairwise comparison, the second primary approach, is considered the gold standard in chart summarization evaluation [3,4,10]. Evaluators view a chart image and two candidate summaries, selecting their preferred summary. However, this method's high cost often constrains study size, failing to account for evaluator variability and chart diversity. Most studies, such as Chart-to-Text and UniChart, also do not specify the preference criteria given to evaluators, obscuring the basis for their judgments and compromising the method's specificity and objective alignment [3, 4].

The challenges in creating reliable and objective-aligned evaluation metrics for chart summarization are similar to those encountered in the more mature field of text

summarization. Text summarization has a longer history and greater resources, which have led to the development of strategies to address these common challenges. In the next section, we will explore how these strategies from text summarization evaluation could be adapted and applied to improve the evaluation of chart summaries.

2.3.2 Inspiration and Opportunities from Text Summarization Evaluation

Text summarization aims to compress long documents into shorter forms that capture the most important information. Two main approaches exist: extractive summarization, which selects and concatenates key sentences from the source text, and abstractive summarization, which generates new sentences that convey the essence of the original document [70]. As chart summarization more closely resembles abstractive summarization, this section focuses on evaluation methods in that domain.

Compared to chart summarization, text summarization benefits from a wealth of large-scale datasets. Notable examples include DUC 2004, containing 500 news articles paired with human-written summaries; CNN/DailyMail, featuring 300,000 articles with author-written highlight summaries; and XSum, a collection of 230,000 articles with single-sentence summaries. These large datasets have first pushed towards the development of automated evaluation metrics like ROUGE, BLEU, CIDEr, and BERTScore, which have subsequently been adopted in chart summarization. However, it was also noted in text summarization

literature that these metrics often poorly align with human judgments of summary quality [69].

To address these limitations, recent text summarization research has shifted towards more informative human evaluation protocols. While they used to only be pair-wise comparison, they now involve expert annotators grading specific quality criteria (QC) on a 5-point Likert scale, as proposed by, later reinforced in the SummEval meta-evaluation of metrics' quality [67, 69]. The four key criteria are:

- Coherence: Assessing the overall structure and logical flow of the summary.
- Fluency: Rating the grammatical correctness and readability of the summary.
- Consistency: Evaluating the factual alignment between the summary and source.
- Relevance: Determining if the summary captures the most important information from the source.

Introducing these evaluation criteria to chart summarization benchmarks has the potential to significantly enhance the informativeness and reliability of quality assessments. While this approach is considered the gold standard in text summarization, it is resource-intensive and currently difficult to automate, as highlighted by both Kryscinski et al. (2019) and Fabbri et al. (2021) [67, 69].

2.3.3 Conclusion

The evaluation of chart summarization methods plays a major role in advancing the field, but current approaches face significant limitations. Heuristic metrics, such as BLEU, ROUGE, and CIDEr, often poorly align with human judgments of summary quality due to their inflexibility and reliance on exact wording. While human evaluator pairwise comparison is often considered the gold standard, it provides limited insight into the specific qualities that influence preferences and fails to capture the nuanced aspects of summary quality that may vary across different types of charts and evaluators.

Drawing inspiration from the more mature field of text summarization, this section explores strategies to address these challenges. Text summarization research has shifted towards more informative human evaluation protocols, involving human annotators grading specific quality criteria on a Likert scale. Adapting these fine-grained criteria to chart summarization has the potential to enhance the reliability and informativeness of quality assessments.

Chapter 3

Design of Large Language Model Agent for Chart Summarization

The field of chart summarization has seen significant advancements in recent years, primarily driven by end-to-end vision models such as Unichart. These models, while demonstrating promising results, rely heavily on extensive labeled datasets comprising chart-summary pairs. The creation and maintenance of such datasets present substantial challenges in terms of cost, time, and scalability, limiting the broader applicability of these approaches in real-world scenarios.

This chapter presents a paradigm for chart summarization that aims to address these limitations. Our proposed method leverages the expansive general knowledge, reasoning capabilities, and tool utilization proficiency of LLMs, combined with the precision and

domain-specificity of chart visual question answering (CVQA) and chart derendering models. This approach forms an autonomous framework centered around an LLM "brain", capable of interpreting and summarizing charts without relying on expensive labeled datasets.

The primary objective of this chapter is to detail the design and architecture of a chart summarization system that circumvents the need for extensive labeled data. In doing so, we address the following research question: How can an LLM-based agent architecture effectively leverage domain-specific models as tools to enable chart summarization without labeled data? Our exploration focuses on a modular architecture comprising a brain module powered by an LLM and specialized tool modules, designed to enable flexible and efficient chart summarization.

The chapter is structured as follows: Section 3.1 provides an overview of the agent architecture and its key components. Section 3.2 presents the detailed implementation of each module, including the brain, action, and perception modules. Section 3.3 concludes with a discussion of the key design aspects, potential strengths and limitations of our approach, setting the stage for the empirical evaluation in the subsequent chapter.

3.1 Architecture Overview

The chart summarization system proposed in this thesis employs an autonomous agent type of architecture, inspired from works such as Prismer and Visual ChatGPT, and following

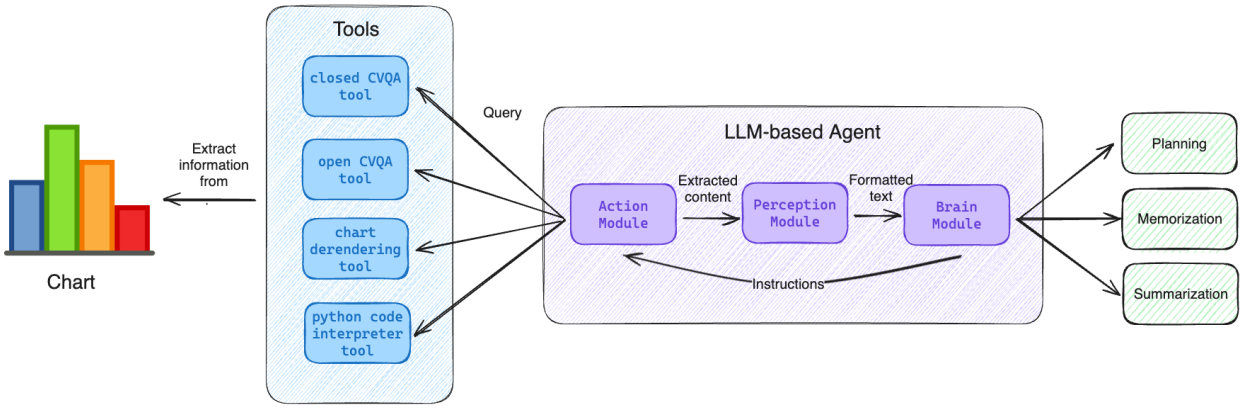


Figure 3.1: Architecture overview of the summarization system

the guidelines established by Xi et al. (2023), which combines the reasoning capabilities of LLMs with the precision of specialized visual processing tools [37, 65, 66]. This approach aims to overcome the limitations of current end-to-end deep learning methods by leveraging a modular, tool-based design that enhances flexibility, improves sample efficiency, and increases interpretability. The architecture of the agent can be observed in Figure 3.1.

The system operates through an iterative process of information extraction, interpretation, and reasoning. The brain module formulates plans and instructions, which the action module executes by interacting with the appropriate tools. The perception module then processes the extracted information, providing the brain module with formatted text for further analysis and decision-making. This cycle continues until sufficient information has been gathered to generate a comprehensive summary of the chart.

This modular design allows for the integration of various specialized tools, such as chart

visual question answering (CVQA) models, chart derendering tools, and code interpreters. These tools can be queried and utilized as needed by the action module under the guidance of the brain module, enabling a flexible and extensible approach to chart analysis.

The design choices for this architecture are motivated by several key rationales. Firstly, this modular tool-based approach offers significant theoretical advantages over traditional end-to-end deep learning methods. Firstly, it enhances flexibility and extensibility by allowing the system to adapt to new tools without the need for retraining. Tools are provided to the model in a few-shot learning style, accompanied by concrete explanations and examples. This adaptability enables the seamless integration of new machine learning models as they emerge, outperforming existing ones in specific domains of chart information retrieval such as label extraction, trend recognition, chart derendering, or CVQA. The section dedicated to the action module will present how and why we selected the specific tools used in our architecture.

Moreover, our approach has the potential to improve sample efficiency and accuracy by leveraging domain-specific deep learning models as tools, thereby reducing the demand for high-quality labeled data typically required to train end-to-end summarization deep learning models. Chart summarization is a complex task that demands strong capabilities in both visual information extraction and language understanding. Traditional approaches would require an extensive dataset of chart-summary pairs, potentially in the order of millions, which is both costly and time-consuming to produce. In contrast, our agent framework

could allow for the use of smaller, more specialized models trained on simpler tasks, such as CVQA and chart derendering, leaving the complex natural language summarization to the large language model. This division of labor potentially leads to improved sample efficiency, accuracy and reduced reliance on large-scale labeled datasets.

Finally, this approach can offer greater interpretability and controllability. Rather than relegating the reasoning and summary generation process to the abstract mechanisms of an end-to-end neural network, we leverage the agent's explicit reasoning capabilities. This allows for a more transparent identification of the information used to generate summaries and how it was interpreted to derive the high-level message conveyed by the chart. Such transparency facilitates easier error diagnosis, model improvement, and potentially, greater trust in the system's outputs.

The subsequent sections will dive deeper into each component of the architecture, delineating their functionalities, interactions, and the rationale behind their design choices. We will explore the implementation details of each module, the selection criteria for the integrated tools, and how these elements cohesively form our chart summarization system.

3.2 Modules Implementation

3.2.1 Perception Module

The primary function of the perception module is to ingest and process information extracted by the action module, preparing it for analysis and interpretation by the brain module. In essence, it serves as an abstraction layer that handles the formatting and post-processing of extracted content before it is passed to the brain module for further reasoning, planning, and memorization.

In the context of chart summarization, the perception module's role is fundamental. It must translate the diverse outputs from various chart analysis tools employed by the action module into a consistent, text-based format that can be readily processed by the LLM serving as the brain module. This translation process comprises the standardization and text formatting of tool outputs. By consolidating and standardizing the heterogeneous outputs from different chart analysis tools, the module ensures a consistent format for the brain module to work with.

The capabilities and limitations of the perception module are inherently tied to the underlying transformer architecture employed in the LLM serving as the brain module. At the time of this thesis's development, most transformer-based LLMs, including those used in our system, did not support direct image ingestion. This limitation necessitated the development of a complex visual information extraction process, handled by the action

module and its associated tools. This workaround allows our system to process chart images indirectly by converting visual information into text-based descriptions that the LLM can understand.

Another significant constraint is the context window limitation of the transformer architecture. LLMs can only attend to a finite number of tokens (subword units) at a time, which effectively limits the amount of text the model can process in a single pass. For most models available during this research, the context window ranged from 8,000 to 16,000 tokens, approximately equivalent to 6,000 to 12,000 words. This constraint poses significant challenges for chart summarization tasks, particularly in terms of the agent's ability to maintain coherent long-term memory, engage in complex multi-step reasoning, and effectively plan the summarization process for intricate charts [71].

To address the context window limitation, we implemented a rolling summarization memory mechanism. This mechanism allows the system to manage scenarios where the information extracted from a chart exceeds the LLM's context window. The details of this mechanism will be elaborated upon in the section dedicated to the brain module and the agent's memory system.

Recent advancements in LLM technology, such as the introduction of GPT-4-Vision, offer complementary capabilities to our current perception module. These multimodal models can directly ingest and process image data, potentially enhancing our architecture's flexibility. Notably, these newer models also boast significantly larger context lengths of up to 128,000

tokens, addressing some of the limitations we've encountered. In the future work section of our discussion, we will explore strategies for integrating these capabilities into our framework, aiming to combine the strengths of both approaches. Additionally, the next chapter will present a comparative analysis of the base capabilities of these newer models against our current system, providing valuable insights into their respective strengths and potential synergies [71].

Having examined the input processing mechanisms of our system, we now turn our attention to its core component: the brain module. This central element of our LLM-based agent architecture is responsible for the tasks of reasoning, decision-making, and summary generation.

3.2.2 Brain Module

At the heart of the brain module lies a LLM, specifically GPT-4 (checkpoint 0613). The choice of GPT-4 was motivated by its superior performance in complex reasoning tasks and its advanced capabilities in "function calling", a feature that allows the model to interact effectively with external tools and APIs. At the time of this work, GPT-4 demonstrated the strongest performance in this area, making it an ideal candidate for our system which relies heavily on tool interaction for chart analysis [72].

3.2.2.1 Prompt Engineering Techniques

To optimize the performance of the LLM in the context of chart summarization, we employed several prompt engineering techniques. Prompt engineering involves crafting specific instructions to induce reasoning, planning, and critical thinking in the LLM [37]. The primary techniques we utilized are chain-of-thought (COT), the ReAct framework, and task decomposition.

Chain-of-thought, introduced by Wei et al. (2023), asks the model to explicitly generate intermediate reasoning steps before accomplishing a task. In our system, this technique is used to break down complex chart analysis tasks into smaller, more manageable steps. For instance, when determining what information to retrieve from a chart, the brain module might reason through identifying the chart type, looking for the title and axis labels, and then examining the data trends. This step-by-step reasoning process has been shown to increase performance by up to 40% on certain benchmarks [47].

Building upon COT, we implemented the ReAct framework proposed by Yao et al. (2023). ReAct interleaves reasoning traces with task-specific actions, which in our case involves querying tools to extract information from charts. This approach significantly enhances the performance of our agent in information retrieval scenarios and reduces hallucination issues commonly encountered with LLMs. Yao et al. (2023) demonstrated that ReAct can outperform direct querying, COT-only, or acting-only methods by up to 30% in similar tasks [60].

To handle the complexity of chart summarization, which involves varying topics, visuals, and data types, we implemented a task decomposition strategy. This approach breaks down complex questions into multiple independent questions. For example, instead of directly tackling "What is the chart about?", the system might ask "What is the title of the chart?", "What are the labels?", "Is there a clear trend?", and "What's the highest value?". This strategy allows the brain module to infer answers to complex questions from the answers to simpler ones, enhancing its reasoning capabilities.

3.2.2.2 Implementation of Reasoning Techniques

The implementation of these techniques in the brain module follows a specific process. When presented with a chart, the brain module uses COT to generate an initial plan for analysis, identifying key aspects of the chart that need to be examined and determining the types of information to be extracted. The initial complex task is then broken down into smaller, more specific questions using the task decomposition strategy.

For each sub-task, the brain module selects an appropriate tool and formulates a query. This selection process is guided by in-context learning (ICL), where the model learns to use tools based on descriptions and examples provided in its prompt [37]. As information is retrieved from the tools, the brain module uses COT reasoning to interpret the results and formulate follow-up questions if necessary.

Following the ReAct framework, the brain module iterates through this process, refining

its understanding of the chart with each cycle. This iterative approach allows for a more thorough and accurate analysis of the chart, as each round of information gathering and reasoning builds upon the insights gained in previous rounds. Once sufficient information has been gathered and processed, the brain module synthesizes a comprehensive summary of the chart.

3.2.2.3 Tool Learning and Utilization

A major aspect of the brain module's functionality is its ability to effectively use a variety of tools for chart analysis. This capability is primarily achieved through ICL, a powerful feature of large language models that allows them to adapt to new tasks based on instructions and examples provided in their input prompt .

In our system, each tool is described to the brain module through carefully crafted prompts. These prompts include a clear description of the tool's functionality, its input parameters, and the expected format of its output. Additionally, we provide several examples of the tool being used correctly in various scenarios. For instance, our "Python Code Interpreter" tool, which is useful for analyzing data from charts, might be described as follows:

"The 'PythonCodeInterpreter' tool executes Python code on a pandas dataframe representation of the chart data. It has access to numpy, pandas, and scipy libraries, as well as a pre-defined 'df' dataframe containing the chart data, obtained from the

'ChartDerendering' tool. A custom 'get_slope' function is available for trend analysis. The tool takes Python code as input and returns the printed output. Example usage: Input: print(df['Value'].mean(), get_slope(df, 'Value', window_size=2)), Output: 42.5, 0.75."

This approach allows the brain module to understand how to use each tool without requiring extensive training or fine-tuning. The LLM's inherent language understanding capabilities enable it to generalize from these examples to novel situations, making it a flexible and adaptable system. In the case of the Python Code Interpreter, this flexibility is particularly valuable as it allows the brain module to perform a wide range of data analysis tasks, from simple statistical calculations to complex trend analyses, adapting its approach based on the specific characteristics of each chart.

When faced with a task that requires tool use, the brain module first assesses which tool is most appropriate based on the current information need. This decision is made by comparing the task at hand with the known capabilities of each tool, as described in their respective prompts. Once a tool is selected, the brain module formulates an appropriate query based on the tool's input requirements.

After receiving output from a tool, the brain module interprets the results in the context of the overall chart analysis task. This interpretation may involve correlating the tool's output with previously gathered information, identifying any discrepancies or unexpected results, and determining if additional information is needed.

3.2.2.4 Memory Mechanism

As mentioned in the section about the perception module, the brain module of our LLM-based agent faces a significant challenge in the form of context length limitations. The GPT-4 checkpoint (0613) employed in our system has a maximum context length of approximately 8,000 tokens (about 6,000 words). This constraint poses a potential issue for our chart summarization task, which often requires multiple rounds of reasoning and tool interactions. As the analysis progresses, there's a risk of "forgetting" valuable information from earlier rounds, potentially compromising the quality and coherence of the final summary.

To address this challenge, we explored several memory management strategies. Initially, we considered selective information retention, which involved filtering the types of information kept in memory. However, our experiments revealed that preserving both reasoning traces and tool calls was central for maintaining coherence between reasoning rounds and enhancing the agent's overall performance. We also explored information compression by using an LLM to summarize reasoning traces, but this approach led to a slight decline in inter-round coherence and significantly increased processing time due to additional LLM calls.

Based on our experimental results, we adopted a rolling window with summarization strategy. This approach offers a balance between memory conservation and information retention. As the agent progresses through reasoning cycles, it accumulates information in its working memory. When the accumulated information approaches the context length

limit (approximately 7,000 tokens in our implementation), the oldest information is summarized into a concise yet insightful paragraph. This summary replaces the original detailed information, freeing up memory space while preserving key insights. The process repeats as necessary throughout the chart analysis task.

This strategy offers several advantages. By summarizing rather than discarding older information, we retain the broad understanding of the chart typically acquired in early analysis stages. The summarization process ensures that the reasoning traces remain coherent and useful throughout the analysis, preventing degradation in later stages. Moreover, it provides memory efficiency by compressing older information, preventing memory overload while still maintaining a comprehensive view of the chart analysis process. The summarization process naturally emphasizes more important or relevant information, as these key points are more likely to be included in the concise summaries.

To illustrate, consider a scenario where the agent analyzes a bar chart showing population growth over time. In early rounds, it identifies the chart type, axes labels, and overall trend. As analysis progresses and it extracts specific data points and performs calculations, earlier rounds are summarized. For instance, the initial chart identification might be condensed to "Bar chart displaying population growth from 1900-2000. X-axis: years, Y-axis: population in millions." This summary replaces the original detailed observations, freeing space for new information while retaining key insights.

In practice, our agent rarely encounters situations where it exceeds the context window, as

most chart analyses can be completed within the available token limit. Still, for particularly complex or data-rich charts that do approach this limit, this approach proves effective. While this method works well in maintaining coherent reasoning and producing high-quality summaries, it inevitably results in some loss of detail through the summarization process.

It is worth noting, however, that newer language models have been developed with significantly longer context windows, largely mitigating the need for such complex memorization strategies. Consequently, while our approach effectively addressed the limitations of the GPT-4 model used in this work, future implementations using more advanced models may not require such memory management techniques.

3.2.2.5 Summary Generation

The summary generation phase is the final step in our chart summarization process. Unlike the initial phase, which focuses on extracting specific information from the chart, this phase tasks the brain module with synthesizing the collected pieces of information into a coherent and insightful narrative.

To facilitate effective summary generation, we employ COT reasoning as a foundational technique. COT allows the brain module to critically assess the meaning of the extracted information and plan the content and structure of the summary. This process involves the model generating an internal dialogue, considering various aspects of the chart data, their relationships, and potential implications. By leveraging COT, the brain module can

make more informed decisions about what information to prioritize and how to present it cohesively.

Building upon this COT foundation, we have developed a specialized prompting strategy distinct from the one used in the information extraction phase. Our approach incorporates several key elements to guide the brain module in producing comprehensive and accurate summaries. Drawing from Kintsch and van Dijk's (1978) model of text comprehension, which emphasizes the importance of condensing and organizing information for effective understanding, we provide specific instructions for the desired summary structure [73]. This typically includes an overview of the chart's main topic, a discussion of key trends or data points, and any notable implications. This structured approach ensures usefulness for all types of users and consistency across various chart types and complexities .

The prompts also include explicit quality criteria as guidelines for the brain module during the generation process. These criteria encompass coherence, consistency, fluency, and relevance. As discussed in our background section, these quality criteria have been identified in other works to correlate well with human preferences for summary quality in text summarization. We believe they are equally relevant to chart summaries and thus apply them here. By explicitly stating these criteria, we aim to guide the LLM towards producing summaries that not only convey factual information but also present it in a clear, well-structured, and meaningful manner.

To further enhance the quality and consistency of generated summaries across various

chart types, we leverage the capability of LLMs to learn from examples, a process known as few-shot learning [52]. We incorporate a set of three manually crafted and carefully reviewed examples of high-quality summaries into our prompts, alongside the previously mentioned instructions and quality criteria. These examples serve as implicit templates, demonstrating the practical application of Kintsch and van Dijk's model of text comprehension and the implementation of our quality criteria. Each example illustrates the desired structure (overview, key data points or trends, implications), tone, and level of detail for different chart types and complexities. By providing these concrete examples, we guide the LLM in producing summaries that not only adhere to our explicit instructions but also emulate the nuanced characteristics of well-crafted summaries.

Cognitive science research supports the efficacy of concise summaries in enhancing comprehension and retention. Miller's (1956) seminal work on working memory capacity suggests that humans can effectively process only a limited amount of information simultaneously [74]. Drawing on these insights, we have implemented a target word count of 50-70 words per summary in our generation process. This constraint serves multiple purposes: it ensures conciseness, aligning with cognitive limitations; it compels the system to prioritize the most salient information; and it maintains consistency across diverse chart types, facilitating user familiarity and expectations. By adhering to these cognitive principles, our approach aims to optimize the balance between informational content and cognitive processing demands, potentially enhancing users' ability to quickly grasp and

retain key insights from the generated summaries.

In situations where the brain module cannot discern a clear overarching message from the extracted information, such as when the data points are highly uncorrelated or contradictory, we employ a specialized prompting strategy. The LLM is instructed to analyze the available data and generate a summary that explicitly acknowledges the uncertainty of its conclusions. For instance, the summary might begin with phrases like "The chart presents varied data points without a clear trend..." or "While the information is inconclusive, possible interpretations include...". This approach serves two important purposes. Firstly, it mitigates the risk of hallucination, a common challenge in LLMs where they generate plausible but false information, by encouraging the model to express uncertainty rather than fabricate connections, we reduce the likelihood of presenting misleading conclusions. Secondly, this strategy is particularly beneficial for BVIP, who have been shown to place high trust in AI-generated content, even when it seems incongruent with the context [75]. By explicitly stating uncertainty in certain situations, we implement a form of negative framing that, according to MacLeod et al. (2017), encourages more appropriate skepticism in BVIP towards AI-generated content. Consequently, this enables BVIP to better interpret the limitations of the AI-generated summary and make more informed decisions based on the available data.

After the initial summary generation, we implement a self-reflection mechanism to further refine the output, inspired by recent research on AI self-improvement [56]. This approach

relies on the capability of LLMs to critically evaluate their own work. We prompt the LLM to assess its generated summary against the previously established quality criteria, encouraging it to identify potential weaknesses or areas for improvement. Based on this self-assessment, we engage in an iterative refinement process, where the LLM revises and regenerates the summary, addressing the identified areas for improvement. Our experimental tests have shown improvement after the first iteration of this process, but diminishing returns thereafter. Considering the trade-off between summary quality and computational efficiency, we have opted to limit the process to a single iteration of self-reflection and refinement.

The brain module's prompting techniques presented form our summary generation strategy. While this approach aims to produce informative and accessible summaries, the quality of the initially extracted information significantly impacts the model's ability to generate insightful content. The action module, which executes the brain's instructions and interfaces with external tools, is therefore crucial to the overall summarization process. The following section will examine the action module's role in supporting effective chart summarization.

3.2.3 Action Module

The action module serves as the third core component of our LLM-based agent architecture for chart summarization, acting as an intermediary between the brain module and the external tools used for information extraction. This section details the design and

implementation of the action module, including its role, the tool interaction protocol, and a comprehensive overview of the integrated tools.

3.2.3.1 Role of the Action Module

The primary function of the action module is to extend the agent's perception capabilities through tool integration. It serves as the interface between the brain module and the machine learning models (tools), handling API requests, error management, load balancing, and parallelization of tool executions. The action module receives instructions from the brain module and executes them by interacting with the appropriate tools. It then passes the extracted information to the Perception Module in a textual format, facilitating the flow of information throughout the system.

During system initialization, the action module provides the brain module with essential information about available tools, including their required parameters and usage instructions. This modular design allows for flexibility and extensibility, as new tools can be integrated without significant changes to other modules. By abstracting the complexities of tool interaction, the action module enables the brain module to focus on high-level reasoning and decision-making tasks.

3.2.3.2 Tool Interaction Protocol

The action module plays an important role in system initialization and ongoing tool interaction. During initialization, it performs health checks on all integrated tools and compiles essential information about each, including their required parameters and usage instructions. This information is then integrated to the prompt of the brain module, enabling informed decision-making about tool selection and utilization throughout the chart analysis process.

To facilitate efficient and reliable communication between the Brain and action modules, we have designed a standardized JSON-based protocol. This protocol defines the structure for the brain module to provide execution instructions to the action module. The JSON schema includes a tool identifier (a unique string specifying which tool to use) and parameters (a dictionary of key-value pairs containing the necessary information for the tool's function). Upon receiving these instructions, the action module is responsible for translating the JSON schema into actual tool calls, extracting and validating the required information before executing the appropriate tool.

For instance, the brain module might provide the following execution instruction for the Unichart ClosedCQA tool:

```
{
    "tool_id": "unichart_closedcqa",
    "parameters": {
        "question": "What is the highest value in the chart?",
        "chart_image": [image_data]
    }
}
```

The action module would then process this instruction, make the appropriate call to the Unichart ClosedCQA tool, and return the result to the brain module. The output might look like this:

```
{
    "status": true,
    "answer": "125 million sales",
    "error_message": null
}
```

Error handling and robustness measures play an important role in ensuring reliable tool interactions. The action module first validates inputs, checking that all required parameters are present and correctly formatted before executing a tool. After execution, it verifies the output by comparing the result against the expected schema, ensuring data integrity. For potential failures, the system uses a retry mechanism with increasing delays between attempts, preventing tool overload. It allows up to three retries within a 30-second window. The action module also maintains detailed logs of all tool interactions, including inputs, outputs, and errors. This comprehensive logging helps in troubleshooting issues and

improving the system over time

To optimize tool execution, the action module employs several techniques. When the brain module provides multiple independent tool execution instructions, the action module uses parallelization to process these instructions simultaneously, reducing overall execution time. The system caches summaries alongside the chart's hash signature, allowing quick retrieval of previously analyzed charts without repeating tool calls. Furthermore, the action module employs batching for multiple independent calls to the same tool, executing them together to improve efficiency. These optimization techniques work in tandem to minimize processing time and enhance the system's overall performance.

3.2.3.3 Tools List

The action module integrates several domain-specific tools, each serving a distinct purpose in the chart analysis process. This section provides a detailed description of each tool, including its capabilities, limitations, and the rationale for its inclusion in the system. To ensure seamless integration, several preprocessing steps are implemented. For the Unichart modes, input images are standardized in terms of size and format to ensure consistent performance. The outputs from these tools are then parsed and normalized into a consistent JSON format, facilitating consumption by the perception and brain modules.

The selection and integration of these tools are guided by three primary factors: the quality of information extracted, the coverage of information that can be obtained, and

the ability to cross-verify information. We prioritize tools that offer high accuracy and reliability in information retrieval while ensuring a diverse range of extractable information through complementary capabilities. Additionally, we maintain some overlap in the type of information extracted by different tools, allowing the brain module to cross-verify and ensure the accuracy and reliability of the extracted information.

The Unichart model, which achieves state-of-the-art performance in CVQA and chart derendering tasks, forms the backbone of our toolset. It provides multiple modes: CVQA, chart derendering, and chart summarization. All these modes share the same self-supervised pretraining checkpoint, meaning they are initially trained on a large dataset without specific task labels, but are then fine-tuned on task-specific datasets to specialize their performance. By utilizing the CVQA and chart derendering modes, we access high-quality information from charts while decomposing the complex chart summarization process into more manageable sub-tasks, each addressed by a dedicated tool.

The Unichart OpenCVQA mode is designed to answer open-ended questions about chart content and relationships. This tool is particularly useful for extracting insights that require understanding basic relationships between different chart elements. While not capable of interpreting the overall message of the chart independently, it can answer a range of open-ended questions about chart content and provide contextual understanding of chart elements. For instance, it can describe the general pattern of a single data series or explain the meaning of specific labels in the context of the chart. However, this mode may

struggle with highly complex or ambiguous charts and has the potential for hallucination, especially when dealing with implicit information. Despite these limitations, the OpenCVQA mode provides a flexible means of extracting insights that complement the more precise, factual information obtained from the ClosedCVQA mode.

Complementing the OpenCVQA mode, the Unichart ClosedCVQA mode is optimized for answering close-ended questions about specific chart values and attributes. Additionally, it can extract labels similarly to an optical character recognition (OCR) model, making it valuable for identifying chart elements. This tool offers high precision for extracting definite facts from charts. It excels at answering specific, well-defined questions about chart data, providing high-confidence responses for quantitative information, and efficiently extracting precise values and relationships from charts. However, this mode has limitations that require careful consideration. For example, when asked to identify the country with the highest value, it might return only one country even if multiple countries share the highest value. Similarly, it may struggle with questions that require understanding context or making inferences beyond the explicitly stated data. Despite these limitations, the ClosedCVQA mode is useful for ensuring the accuracy of quantitative statements in the generated summaries and for verifying specific data points.

The Unichart Chart Derendering mode serves an important function in enabling detailed data analysis and manipulation. This tool is responsible for converting the visual representation of data in charts into a structured, tabular format. It can extract raw data

values and relationships from a wide range of chart types, convert visual data into a structured table format, and preserve relationships between different data series in multi-series charts. To enhance its capabilities, we have implemented a post-processing step using a LLM. This additional step was introduced to clean the table and create a consistent format across all chart types, addressing the variability in output formats that the original model produced for different chart types. The LLM-based post-processing includes cleaning empty values and performing appropriate value conversions to make the output compatible with Python and Pandas libraries. Since we typically need to extract the data table for each chart, we precompute the chart data table at the initialization of the system, allowing the Brain Module to access this information through the Python Pandas Interpreter tool without delay. While this mode may struggle with highly unusual or complex chart designs and has the potential for transcription errors, especially with dense or overlapping data points, it provides the raw data necessary for in-depth analysis and verification of chart information.

To enhance the system's flexibility and extensibility, we have integrated a Python Pandas Interpreter tool. This tool allows the Brain Module to execute custom Python code on the dataframe generated by the chart derendering tool. It can utilize popular data science libraries such as Pandas and NumPy, enabling the computation of advanced statistics and the generation of derived insights. We have implemented a custom function that fits a linear model to two series and returns the fitting score as well as the coefficient of proportionality.

This function addresses a common need of the Brain Module in interpreting the significance of the chart, ensuring robustness in the implementation. The main challenges with this tool include the need for careful prompt engineering to ensure relevant and safe code generation, the potential for errors or inefficiencies in generated code, and limitations imposed by the computational resources allocated to the interpreter.

To facilitate integration and ensure modularity, all tools were implemented as Python applications with HTTP endpoints, containerized using Docker. This approach provides a standardized interface for tool interactions, ensures easy reproducibility, and allows for potential scalability. The models were served on a graphics processing unit (GPU) with 8GB of VRAM, providing sufficient computational power for the processing of chart images and related queries.

3.3 Discussion

This section synthesizes the key design choices of our proposed chart summarization system, analyzes its potential strengths and limitations, and sets our expectations for the empirical evaluation that follows.

3.3.1 Synthesis of Key Design Choices

The cornerstone of our approach is a modular LLM-based agent architecture, comprising three primary components: the brain, action, and perception modules. The brain module,

powered by a large language model (specifically GPT-4), serves as the central reasoning and decision-making unit. It formulates plans for information extraction, interprets the gathered data, and generates the final summary. The action module acts as an intermediary, translating the brain's high-level instructions into specific tool interactions. Finally, the perception module processes and formats the information extracted by the tools, preparing it for the brain's consumption.

This modular approach offers several advantages in addressing key chart summarization challenges. Primarily, it enables chart summarization without relying on extensive labeled datasets of chart-summary pairs. Instead, our system leverages the general knowledge and reasoning capabilities of LLMs, combined with the specific strengths of domain-specific tools for chart analysis. This design choice is motivated by the scarcity and high cost of obtaining large-scale, high-quality labeled datasets for chart summarization.

Furthermore, our approach theoretically improves sample efficiency and flexibility. By utilizing pre-trained LLMs and specialized tools, the system can potentially generalize to a wide range of chart types and domains with minimal additional training. The modular nature also allows for easy integration of new tools or replacement of existing ones as better models become available, enhancing the system's adaptability to evolving requirements.

However, the use of AI in chart summarization raises important ethical considerations. Potential biases in the underlying LLMs or domain-specific tools could lead to skewed or inaccurate summaries. To address this, we have instructed the brain module to be cautious

in its interpretation of the gathered data. It is programmed to note any suspicions of inconsistency or errors in the final summary, thereby raising awareness for users, particularly those who are visually impaired. This approach aims to improve accessibility while ensuring that the generated summaries effectively convey the chart's visual information without introducing new barriers or misinterpretations.

3.3.2 Strengths and Weaknesses of the Proposed Approach

The LLM-based agent approach offers several theoretical strengths. Its potential for improved generalization stems from the broad knowledge base of LLMs, which can be applied to understand and describe various chart types and topics. Moreover, the strong language capabilities of LLMs enable the generation of fluent and well-articulated summaries, a key aspect of our task. The combination of general LLM capabilities with specialized tools allows for a division of labor that leverages the strengths of each component. For instance, while the LLM excels at natural language understanding and generation, the specialized tools can perform precise data extraction and analysis tasks.

However, this approach also presents potential limitations and challenges. One significant challenge lies in coordinating multiple components and ensuring consistent information flow between them, which requires careful system design and integration. Each tool introduces its own set of limitations and potential biases, which must be carefully managed. Moreover, the effectiveness of the system heavily relies on the brain module's ability to select and

utilize the appropriate tools for each chart, a task that may prove challenging for complex or unconventional chart types. However, LLMs have shown an ability to reconcile potentially inconsistent information from various sources, which is instrumental for making sense of data gathered from different tools.

When compared to end-to-end approaches like Unichart and VLLMs like GPT-4, our modular system offers distinct trade-offs. End-to-end models potentially provide more seamless and implicit integration between visual understanding and language generation but often require large amounts of task-specific training data. VLLMs, while more powerful, require even more extensive datasets and are typically not accessible for open-source development due to their size and computational requirements. Our approach, while potentially more complex in terms of system design, offers greater flexibility and interpretability. It allows for targeted improvements of individual components and provides a clearer path for diagnosing and addressing issues in the summarization process.

3.3.3 Conclusion

In conclusion, our proposed chart summarization system represents an alternative approach to addressing the challenges of chart summarization without extensive labeled data. By leveraging the strengths of LLMs and specialized tools in a modular architecture, we aim to create a flexible and adaptable system capable of generating high-quality summaries across diverse chart types and domains.

Future research directions could explore the integration of more advanced LLMs or multi-modal models as they become available, potentially enhancing the system's ability to understand and describe visual data. Additionally, the exploration of additional specialized tools could further expand the system's capabilities in handling complex or domain-specific charts.

While our theoretical analysis suggests potential advantages of our approach, empirical validation is necessary to confirm these benefits. The next chapter will evaluate our system's performance, focusing on summary quality and its ability to address the research questions posed at the beginning of this thesis.

Chapter 4

System Evaluation

Building upon the foundation laid in Chapter 3, where we introduced a novel LLM-based agent architecture for chart summarization, this chapter aims to provide a thorough evaluation of our proposed system against relevant baselines, namely Unichart and GPT-4-Vision. The evaluation process is designed not only to gauge the performance of our system but also to contribute to the broader discourse on effective assessment methodologies in the field of chart summarization.

Traditional evaluation approaches in chart summarization face significant limitations. Automated metrics like BLEU and ROUGE, while widely used, rely on expensive labeled datasets and often fail to capture the subtle aspects of summary quality, particularly in visual-textual contexts. Human evaluator rankings, though valuable, are resource-intensive, limited in scale, and provide little insight into the reasons behind preferences. To address

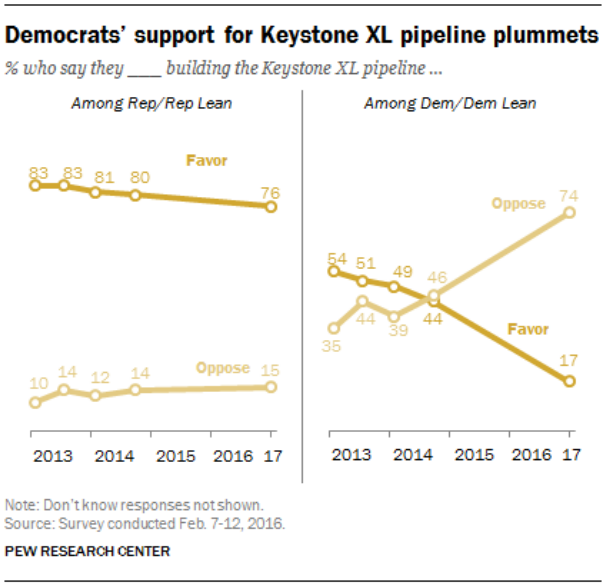
these challenges, we adapt quality criteria from text summarization to provide a more informative and multifaceted assessment of chart summaries. This adaptation allows us to precisely characterize chart summaries and investigate the key criteria influencing user satisfaction across different user groups.

In this chapter, we aim to address two research questions that are central to our evaluation of chart summarization systems and methodologies. First, we seek to determine how our modular LLM-based agent approach compares to end-to-end visual-language training and VLLMs in terms of summarization quality. Second, we seek to explore the key factors that influence user satisfaction across different groups, as assessed by our adapted quality criteria.

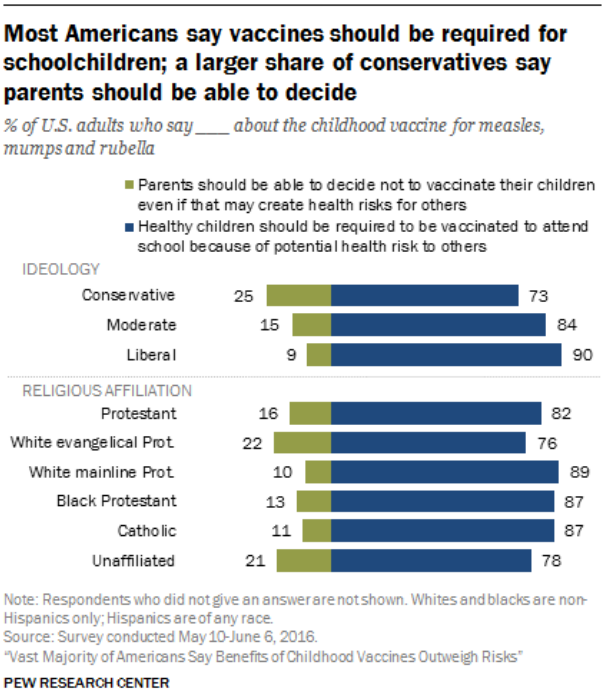
The remainder of this chapter provides an evaluation of our chart summarization system, addressing the aforementioned research questions through a structured approach. We begin by examining the dataset used for evaluation in Section 4.1, focusing on its characteristics and relevance to real-world scenarios. Section 4.2 then outlines the baseline methods selected for comparison, justifying their inclusion based on performance and relevance. Our evaluation methodology is detailed in Sections 4.3 and 4.4. The former introduces the quality criteria framework adapted for this study, explaining each criterion and its application. The latter describes the design and implementation of our user study, encompassing participant selection, materials preparation, and data collection procedures. Section 4.5 presents an analysis of the evaluation results, examining both system performance and user preferences. The chapter concludes with Section 4.6, which discusses

the implications of our findings, acknowledges limitations, and proposes directions for future research in chart summarization evaluation.

4.1 Dataset



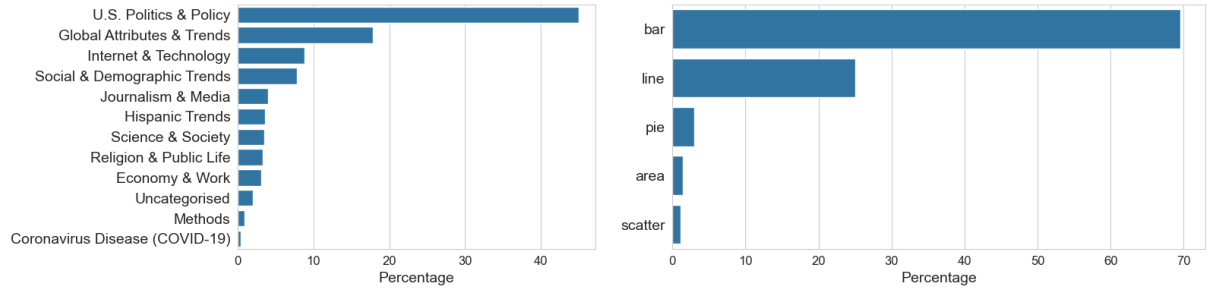
(a) Democrats' support for Keystone XL pipeline over time



(b) Opinions on vaccine requirements for schoolchildren

Figure 4.1: Examples of Charts from Pew Chart2Text Subset

For the evaluation of our chart summarization system, we utilized the Pew subset of the Chart-to-Text dataset [4]. This choice was motivated by several factors that align with our research objectives and the real-world applicability of our proposed LLM-based AI agent approach.



(a) Topics in the Pew Chart2Text subset

(b) Chart types in the Pew Chart2Text subset

Figure 4.2: Distributions of topics and chart types in the Pew subset of Chart2Text dataset

The Pew subset comprises 10,000 real-world charts extracted from articles published by the Pew Research Center. For our study, we focused specifically on the 1,200 charts from the test split, ensuring a robust evaluation of our system on unseen data. This dataset offers a diverse and representative sample of charts encountered in actual research and journalism contexts, making it particularly suitable for assessing the performance of our chart summarization system in practical scenarios.

One of the key strengths of this dataset is its topical diversity. The charts span 12 distinct categories, covering a wide range of subjects typically addressed in social and economic research. This variety is crucial for evaluating our system's ability to generate accurate and contextually appropriate summaries across different domains, a capability that is essential for a versatile chart summarization tool. In terms of chart types, the dataset primarily consists of bar and line plots, which together account for over 90% of the total number of charts. Additionally, it includes pie charts, area plots, and scatter plots, albeit in smaller proportions. This distribution reflects the prevalence of different chart types in real-world publications

and allows us to assess our system's performance across various visual representations of data.

Each chart in the dataset is accompanied by rich metadata, including the chart title, chart type classification, bounding boxes around main visual elements, and the underlying data table of values represented in the chart. While the dataset also includes captions for each chart, previous research has identified significant limitations in their quality. These captions, generated through automated heuristics that selected relevant surrounding paragraphs, often miss important information or include details not present in the chart itself, potentially leading to serious hallucination problems for models trained on these captions [4]. Given these limitations, we have made a deliberate decision to exclude these captions from our evaluation process. Instead, our research will focus exclusively on the chart images and the aforementioned metadata annotations. This approach allows us to leverage the most reliable components of the dataset, ensuring that our evaluation is based on accurate information.

The selection of this dataset and its components allows us to assess our chart summarization system's performance in a context that closely mirrors real-world scenarios. With this robust dataset in place, we can now turn our attention to establishing appropriate evaluation baselines, which will serve as crucial reference points for measuring the effectiveness of our proposed approach.

4.2 Baselines

In order to evaluate the effectiveness of our proposed chart summarization system, we have selected two state-of-the-art models as baselines for comparison: Unichart and GPT-4V.

UniChart is an end-to-end visual-language model specifically designed for chart understanding and summarization. It was trained on a large, diverse dataset of over 627,000 charts using a multi-stage learning approach. The model's architecture consists of a visual encoder to process chart images and a text decoder to generate outputs. UniChart's training procedure involved an initial pretraining phase on multiple chart-related tasks, including data table generation, numerical and visual reasoning, open-ended question answering, and chart summarization. This comprehensive pretraining strategy aimed to imbue the model with a broad understanding of chart elements and their relationships. Following pretraining, UniChart was fine-tuned to create three specialized models, each focusing on a different aspect of chart understanding: chart visual question answering, chart derendering, and chart summarization. These models were fine-tuned on a public benchmark dataset, specifically the Chart-to-Text corpus, which includes charts from sources like Statista and Pew Research Center. This fine-tuning phase allowed each model to specialize in its respective task while building upon the foundation established during pretraining. This model held the state-of-the-art position in chart summarization until late 2023 [3].

We chose Unichart as our first baseline due to its direct relevance to the chart

summarization task and its impressive performance. However, it is important to note that Unichart's reliance on a large, labeled dataset of chart-summary pairs highlights one of the key motivations for our research: to develop a system that can perform well without the need for extensive, task-specific labeled data.

The second baseline we selected is GPT-4V, a large-scale pretrained VLLM by OpenAI. Released in late 2023, GPT-4V extends the capabilities of the GPT-4 language model to include visual understanding. While the exact details of its training procedure are not publicly available, it is reasonable to assume that GPT-4V was trained on a vast corpus of text and images, likely largely surpassing the scale of Unichart's training data [58].

GPT-4V was selected as a baseline due to its state-of-the-art performance in visual-language tasks and its potential suitability for chart summarization. Given the scale of its pretraining, it likely encountered numerous chart images, and considering the base GPT-4 model's proficiency in text summarization, we can reasonably infer GPT-4V's capability in chart summarization. This combination of visual understanding and presumed summarization skills makes GPT-4V a compelling baseline, despite the lack of transparency regarding its specific training on chart-related tasks.

By comparing our proposed system against both Unichart and GPT-4V, we aim to evaluate its performance from different perspectives. Unichart provides a comparison against a specialized, end-to-end supervised model, while GPT-4V offers a benchmark for performance against a general-purpose, large-scale VLLM. This dual comparison will help

us assess the effectiveness of our approach in leveraging AI agents with domain-specific tools for chart summarization.

While these baselines offer strong points of comparison, effectively evaluating chart summarization models requires a tailored approach. Traditional text summarization metrics may not fully capture the nuances of chart summarization, where visual elements play a crucial role. To address this, we adapt an existing quality criteria evaluation framework from the field of text summarization to the context of charts. In the following section, we present this adapted framework.

4.3 Quality Criteria Evaluation Framework

The evaluation of chart summarization presents unique challenges that require a specialized framework. While text summarization has established evaluation protocols, chart summarization methods are comparatively underdeveloped. This section presents a quality criteria evaluation framework for assessing chart summaries.

The motivation for adapting text summarization criteria to chart summarization stems from several factors. Traditional metrics such as BLEU and ROUGE have shown poor correlation with human judgment of summary quality in the context of charts [69]. These metrics often fail to capture details that significantly impact summary quality [76]. Moreover, their reliance on labeled data, which is scarce and expensive to obtain for chart summarization, limits their scalability and applicability in this domain.

While pairwise comparison of summaries by human evaluators is considered the gold standard in chart summarization evaluation due to its reliability, it lacks granularity in explaining preferences and characterizing summary styles. To address these limitations, we propose adapting the quality criteria framework from text summarization, as defined by Kryscinski et al. (2019), to the context of chart summarization [67].

The framework comprises four key criteria: coherence, fluency, consistency, and relevance. In the context of chart summarization, these criteria are defined as follows:

- Coherence: Collective quality of the sentences made the overall structure and logical flow of the summary.
- Fluency: Quality of individual sentences as in the grammatical correctness and readability of the summary.
- Consistency: Factual alignment between the summary and the source.
- Relevance: Selection of the most important information from the source.

To adapt this framework for chart summarization, we simply redefine the "source" as the chart image, rather than the text documents typically used in traditional summarization tasks.

Fabbri et al. (2021) established a protocol for applying the quality criteria framework in text summarization evaluation [69]. Their process began with the random selection of 100 articles from the CNN/DailyMail test set, a widely used benchmark dataset for text summarization consisting of news articles paired with human-written summaries, ensuring

a diverse sample. For evaluator recruitment, they adopted a dual approach: engaging crowd-sourced annotators with professional English proficiency and a proven annotation track record, while also involving three expert annotators with academic experience in summarization. This strategy balanced scalability with expertise. The evaluation was conducted on an online platform where annotators rated computer-generated summaries on a 5-point Likert scale for each quality criterion, with each summary assessed by eight different evaluators to ensure reliability. Their analysis utilized Krippendorff's alpha coefficient to measure inter-annotator agreement and computed correlation scores between human ratings and automated metrics. This methodology revealed that while most automated metrics correlated poorly with human judgment, the proposed criteria showed high correlation with human preference, were interpretable, and measured distinct aspects of summary quality. By averaging scores for each criterion, Fabbri et al. compared model performance across these dimensions, demonstrating the framework's utility for evaluation purposes.

The quality criteria framework, adapted from text summarization, provides a structured approach to evaluating chart summaries. The following section describes the design of our user study, which employs this framework to compare the performance of chart summarization models.

4.4 User Study

To evaluate the effectiveness of our proposed LLM-based AI agent approach for chart summarization, we conducted a user study with three primary objectives. First, we aimed to collect quantitative data on the quality of model outputs using the adapted quality criteria framework described in Section 4.3. This assessment would provide insights into how well each model performs across the dimensions of coherence, fluency, consistency, and relevance. Second, we sought to gather data on user preferences by having participants rank summaries from different models. By comparing these rankings with the quality criteria ratings, we could identify which factors most strongly influence user satisfaction. Finally, we collected demographic and expertise data to investigate whether domain knowledge impacts user preferences and quality assessments of chart summaries.

Ethical considerations were important in the design and execution of our study. We obtained approval from the university's research ethics board (REB) prior to commencing the study. Participants were provided with clear information about the study's purpose, duration, and the nature of their involvement, and were offered a compensation of $15 for their participation. Informed consent was obtained from all participants, and they were assured of their right to withdraw at any time without penalty. To protect participant privacy, all data collected was anonymized and stored securely. Participants were informed that the study carried risks similar to those involved in participating in any video conference or regular computer use, including potential discomfort when discussing feelings related to

technology use.

4.4.1 Participant Selection and Categorization

An important part of our study design was the comparison of responses between expert and novice users in the domain of the charts being summarized. This focus was motivated by previous research in text summarization, which suggests that experts and novices may have differing preferences and needs when it comes to summary content and style [77]. We hypothesized that similar differences might exist in the context of chart summarization. Understanding these potential differences is valuable for the development of summarization systems that can effectively serve diverse user groups.

Given the range of topics covered in the Pew Research Center Dataset, we selected "Economy & Work" as our primary domain of study. This choice was driven by several factors. First, the field of economics often requires specialized terminology, understanding of complex relationships between multiple variables, and substantial background knowledge, making it an ideal domain to distinguish between expert and novice users. Second, this topic offered sufficient diversity in chart types, allowing us to evaluate our system's performance across various visual representations of data. Lastly, economic data is often complex and intricate, presenting a challenging test case for our summarization system.

To recruit participants with varying levels of expertise, we targeted different university departments. For potential experts, we reached out to the departments of Economics,

Management in Data Analytics, Finance, and Politics. To recruit novices, we approached faculties less likely to have extensive economics knowledge, such as the Faculty of Arts in which subjects of literature, art, and philosophy are taught and studied. This strategy aimed to ensure a diverse pool of participants with a range of domain knowledge.

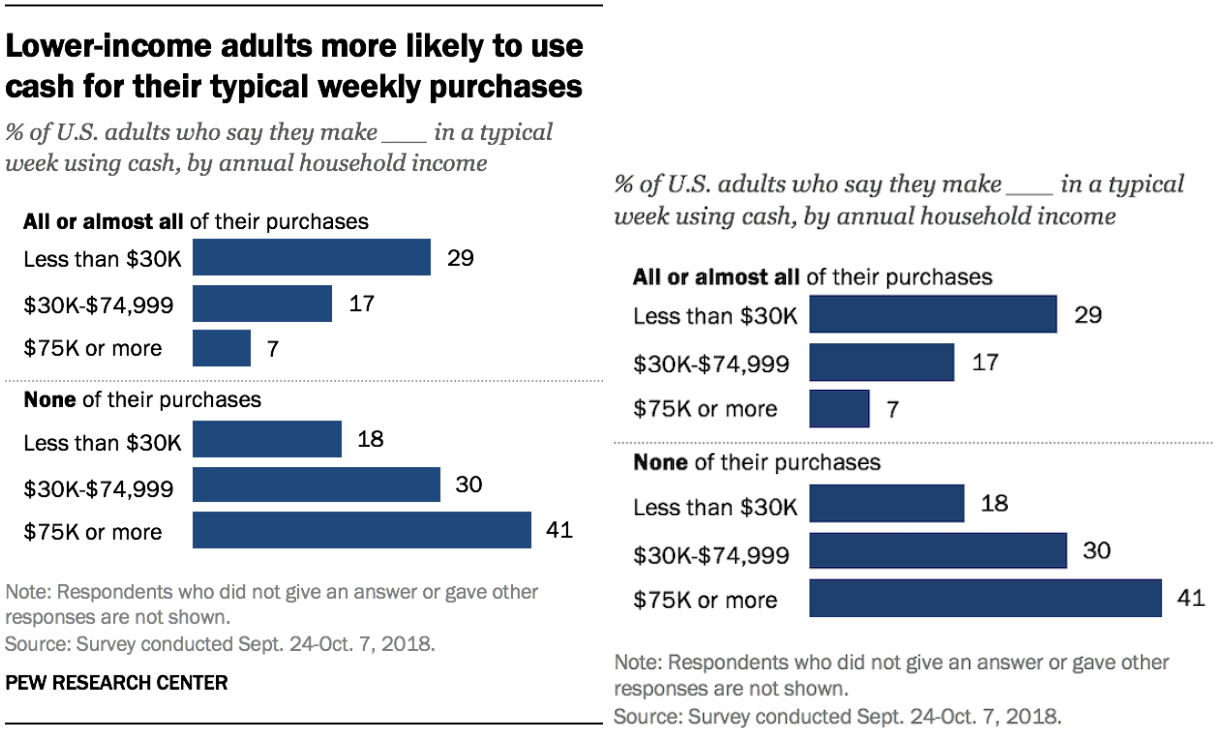
To objectively classify participants as experts or novices, we developed a knowledge questionnaire comprising 15 multiple-choice questions of varying difficulty levels: four easy, seven medium, and four hard. For each question, participants had to choose one answer among four possibilities. The questionnaire was designed to ensure its validity in assessing economic knowledge. To discourage random guessing on harder questions, which reward more points, we included an "I don't know" option for the hard questions only. We did not offer this option for medium and easy questions to encourage participants to think through these questions rather than immediately selecting "I don't know" when unsure. Correct answers were awarded points based on their difficulty: one point for easy questions, two points for medium questions, and three points for hard questions. The questionnaire's difficulty levels were verified and calibrated with input from a doctoral student in the Economics department. The expertise score was calculated using the following formula:

$$\text{Expertise Score} = \frac{e \times 1 + m \times 2 + h \times 3}{4 + 7 \times 2 + 4 \times 3} \times 10$$

Where e, m, and h represent the number of correct answers in the easy, medium, and hard categories, respectively. This scoring system yields a range from zero (minimal expertise) to

ten (maximum expertise). We classified participants as novices if they scored four or below, experts if they scored seven or higher, and intermediate for scores between four and seven.

4.4.2 Materials Preparation



(a) Original chart image. **(b)** Modified chart image.

Figure 4.3: Comparison of original and modified charts.

The preparation of materials for this study involved three primary components: the selection and modification of charts, the generation of chart summaries using different systems, and the compilation of these elements into the evaluation form.

For this study, we selected ten charts from the Pew Research Center test set, specifically

from the "Economy & Work" category. The selection comprised four bar charts and six line charts, presenting a variety of styles and content. This mix was intended to provide a comprehensive assessment of the summarization models' capabilities across different chart types commonly used in economic reporting.

The decision to limit the study to ten charts was based on considerations of participant fatigue and time constraints. Our preliminary assessments indicated that evaluating and ranking three summaries for each chart would be a cognitively demanding task. We determined that ten charts would strike an optimal balance between gathering sufficient data and maintaining the quality of participant responses, while keeping the total study completion time to approximately one hour per participant.

Each selected chart underwent minimal modifications. We only cropped the original titles from the charts, as these often explicitly stated the main message of the visualization. This decision was motivated by two factors: firstly, it more closely simulated real-world scenarios where charts may not always have clear, descriptive titles; secondly, it presented a more challenging task for the summarization systems, requiring them to derive the main message solely from the visual elements and data presented in the chart. Figure 4.3 illustrates this preprocessing step, showing an example of an original chart and its cropped version.

Following the chart preparation, we generated summaries using the three systems described in Section 4.2: Unichart, GPT-4-Vision, and our proposed LLM-based summarization agent. The process for each model was as follows:

- Unichart: This neural network model received the cropped charts as direct input and generated summaries based on its training.
- GPT-4-Vision: We provided this model with the same prompt designed for the summary generation phase of our agent (as detailed in Section 3.2.2.5).
- LLM-based Summarization Agent: Our proposed model processed the cropped charts following the multi-step approach described in Chapter 3, which involves information extraction using domain-specific tools followed by summary generation using a LLM.

The prepared charts and their corresponding summaries were then assembled into an online Microsoft Form. This form was structured in two main parts to facilitate a comprehensive evaluation process. The first part began with a brief demographic questionnaire, designed to assess participants' familiarity with data visualizations and their primary sources of knowledge about economic topics. The expertise assessment quiz, as described in Section 4.4.1, was also integrated into this introductory section of the form.

The second part of the form focused on the evaluation of chart summaries. This section presented the chart images alongside their corresponding summaries for preference ranking and quality criteria rating. The specific layout and methodology for this evaluation process will be elaborated in the following section on data collection procedures.

4.4.3 Data Collection Procedures

Prior to the evaluation, participants received a detailed email outlining the study procedures. This communication served to prepare them for the task and ensure consistent understanding across all evaluators. The email informed participants about the study's purpose of improving access to visual data for visually impaired individuals through automated chart summarization. It also set clear time expectations, advising that the evaluation would take approximately one hour, with 10-15 minutes allocated for the initial questionnaire and 45-50 minutes for the main evaluation of chart summaries.

An important component of the preparation was the introduction of the four quality criteria: coherence, fluency, consistency, and relevance. Each criterion was presented with its definition and accompanying guiding questions for participants to consider during their evaluations. To further enhance understanding of the grading process, participants were provided with a sample chart image and examples of poor summaries for each quality criterion. This approach was chosen to calibrate participants' expectations and ensure more consistent evaluations across the study. The email also provided instructions for participants to rank the summaries based on their personal preference.

To minimize external variables and ensure focus, participants were asked to complete the evaluation in a quiet setting and in one sitting. Contact details for technical or procedural support were provided, along with information about the $15 compensation upon completion, acknowledging participants' time and effort.

The evaluation process was structured to collect both quantitative ratings and qualitative feedback for each chart summary, while minimizing potential biases. To avoid order effects, all charts were presented to participants in a randomized sequence. For each chart, participants were first shown the image and instructed to thoroughly understand its main message before proceeding. This step ensured that evaluators had a solid grasp of the chart's content, enabling more accurate assessments of the summaries.

Following this, all summaries for a given chart were presented simultaneously in a randomized order, without identifying which model generated each summary. This blind presentation method was chosen to minimize bias and encourage objective comparisons between summaries. To ensure consistent evaluation across all charts and summaries, participants were reminded of the quality criteria definitions before each assessment.

The quantitative assessment consisted of two components. First, participants rated each summary on a 5-point Likert scale (1 = poor, 5 = excellent) for each of the four quality criteria. In addition, after completing the individual ratings, participants ranked the summaries in order of preference. To capture the reasoning behind these rankings, participants were then invited to optionally leave a comment explaining their thought process or sharing any observations about the summaries themselves.

This process was repeated for all charts in the study. Upon completion of the entire evaluation, participants were thanked for their time and valuable input. To facilitate a more in-depth exploration of participants' experiences, they were asked if they would be willing

to participate in a follow-up interview. However, none of the target participants, expressed willingness to engage in these interviews, so they did not take place. Finally, participants were given the opportunity to leave a comment about their overall experience with the study or share any additional thoughts they had.

Section 4.5 presents an analysis of the collected data. This analysis encompasses the performance evaluation of different models across the established quality criteria and an investigation of user preferences.

4.5 Results

This section presents the outcomes of our user study on chart summarization, focusing on the performance of our proposed LLM Agent compared to two baseline models: GPT-4-Vision and Unichart.

Our study involved twenty participants: two novices, twelve intermediates, and six experts in fields related to economics and data analysis, as determined by the expertise questionnaire. The slight skew towards higher expertise levels can be attributed to the fact that many respondents were graduates from fields targeted for expert recruitment. The gender distribution was relatively balanced, with eleven men and nine women. Each participant evaluated summaries for ten different charts, each generated by three distinct models. This process yielded a total of 600 individual summary evaluations, providing a comprehensive dataset for analysis.

In the following subsections, we examine the user preferences (Section 4.5.1) and quality criteria (Section 4.5.2) in detail. These analyses offer insights into model performance across various dimensions of summary quality and user satisfaction, considering the impact of participant expertise levels on these assessments.

4.5.1 User Preferences

As part of our comprehensive evaluation, we analyzed user preferences to gain insights into how participants ranked the different chart summarization systems.

Figure 4.4 presents the distribution of rankings for each model across all charts and participants. The bar chart illustrates the number of times each model (GPT-4V, LLM Agent, and Unichart) was ranked first, second, or third in the evaluation process.

As observed on the figure, GPT-4V consistently received the highest number of first-place rankings, indicating a strong overall preference among participants. LLM Agent predominantly received second-place rankings, while Unichart was most frequently ranked third.

To quantify the level of agreement among raters and assess the statistical significance of these preferences, we employed several statistical tests. First, we calculated Kendall's W coefficient to measure the degree of agreement among raters across all scenarios. The average Kendall's W value was 0.3317, indicating a moderate level of agreement among participants. This suggests that while there is some consensus in the rankings, there is also a degree of

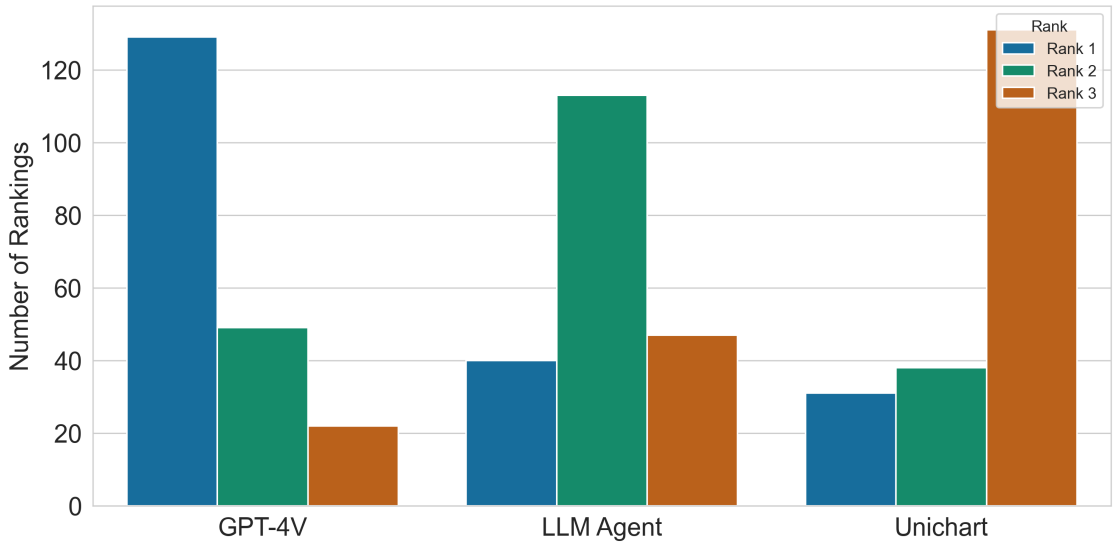


Figure 4.4: Ranking Distribution of Chart Summarization Models Across Evaluation Criteria. This figure illustrates how often GPT-4V, LLM Agent, and Unichart achieved each rank (1st, 2nd, or 3rd) in the evaluation. GPT-4V most frequently achieves the first rank, the LLM Agent predominantly occupies the second rank, and Unichart is most often placed third.

variability in individual preferences.

To determine whether the observed differences in rankings were statistically significant, we conducted a Friedman test. The test yielded a statistic of 107.49 with a p-value of 10^{-24}, which is well below the conventional significance level of 0.05. This result provides strong evidence of statistically significant differences in the rankings of the three models.

Given the significant result of the Friedman test, we proceeded with a post-hoc analysis to identify specific pairwise differences between models. We employed the Nemenyi post-hoc test, which is specifically designed for use after a Friedman test and provides a conservative

Comparison	p-value	Significant
GPT-4V vs. LLM Agent	10^{-4}	Yes
GPT-4V vs. Unichart	10^{-4}	Yes
LLM Agent vs. Unichart	10^{-4}	Yes

Table 4.1: Nemenyi Post-hoc Test Results

approach to control for family-wise error rate in multiple comparisons. The results of the Nemenyi test are presented in Table 4.1.

The Nemenyi test results reveal that all pairwise comparisons between models show significant differences ($p < 0.05$). This finding corroborates the visual interpretation of Figure 4.4, confirming that the ranking distributions for all three models are statistically different from each other.

To further investigate potential differences in preferences between experts and novices in the domain of "Economy & Work", we employed an aligned rank transform (ART) ANOVA. This method was chosen for its ability to handle non-parametric data in factorial designs, making it particularly suitable for our ranking data. The ART ANOVA allows us to examine main effects and interactions while accounting for the within-subject nature of our design across different scenarios. Table 4.2 presents a comprehensive summary of the ART ANOVA results, including all main effects and interactions tested in our study.

The analysis revealed a highly significant main effect of model ($F(2, 227) = 39.72, p < 0.001$), confirming our earlier findings from the Friedman test. Interestingly, we found no significant main effect of Group (expert vs. novice) ($F(1, 227) = 0, p = 1.000$), indicating that

Factor/Interaction	Test Statistic	p-value	Significant
Model	F(2, 227) = 39.72	10^{-15}	Yes
Group (expert vs. novice)	F(1, 227) = 0	1.0	No
Scenario	F(9, 227) = 0	1.0	No
Group × Model	F(2, 227) = 35.59	10^{-14}	Yes
Group × Scenario	F(9, 227) = 0	1.0	No
Model × Scenario	F(18, 227) = 0.01	1.0	No

Table 4.2: ART ANOVA results showing significant effects for Model and Group × Model interaction, with no significant effects for other factors or interactions. This indicates preference differences between experts and novices across summarization models, consistent across scenarios.

overall, experts and novices did not differ in their rankings across all models and scenarios. However, a significant Group × Model interaction was observed ($F(2, 227) = 35.59, p < 0.001$), suggesting that the ranking patterns for the three models differ between experts and novices.

To better understand this interaction, we conducted a detailed post-hoc analysis using Mann-Whitney U tests with Bonferroni correction for multiple comparisons. Table 4.3 presents the mean ranks and effect sizes for each model, comparing experts and novices.

While none of the individual comparisons between experts and novices for each model reached statistical significance after Bonferroni correction, the effect sizes suggest differences in how LLM Agent and Unichart are perceived by the two groups. Specifically, GPT-4V was consistently ranked highest by both experts and novices, with very little difference between the groups (effect size $r = 0.0335$). LLM Agent tended to be ranked lower by novices

Model	Mean Rank		Effect Size (r)	Adjusted p-value	Significant
	Expert	Novice			
GPT-4V	1.45	1.50	0.0335	0.7289	No
LLM Agent	1.98	2.30	0.1826	0.0720	No
Unichart	2.57	2.20	0.1640	0.0870	No

Table 4.3: Comparison of model rankings between experts and novices. Effect sizes (r) indicate the magnitude of difference between groups, with values closer to 0 suggesting smaller differences. Adjusted p-values are after Bonferroni correction.

compared to experts (mean ranks 2.30 vs. 1.98, effect size $r = 0.1826$). Unichart tended to be ranked lower by experts compared to novices (mean ranks 2.57 vs. 2.20, effect size $r = 0.1640$).

The lack of a significant scenario effect ($F(9, 227) = 0, p = 1.000$) in our ART ANOVA indicates that the performance of the models and the differences between expert and novice ratings were consistent across different types of charts. This suggests that the observed preferences are not dependent on specific chart types.

In conclusion, our analysis reveals differences in the perceived quality of chart summaries generated by different models, with GPT-4V consistently receiving the highest rankings across both experts and novices. The expertise level of the evaluators appears to influence the perception of LLM Agent and Unichart, but not GPT-4V, although these differences did not reach statistical significance in our post-hoc tests. Specifically, participants considered experts in the field of "Economy & Work" tended to rank LLM Agent higher, while novices tended to rank Unichart higher.

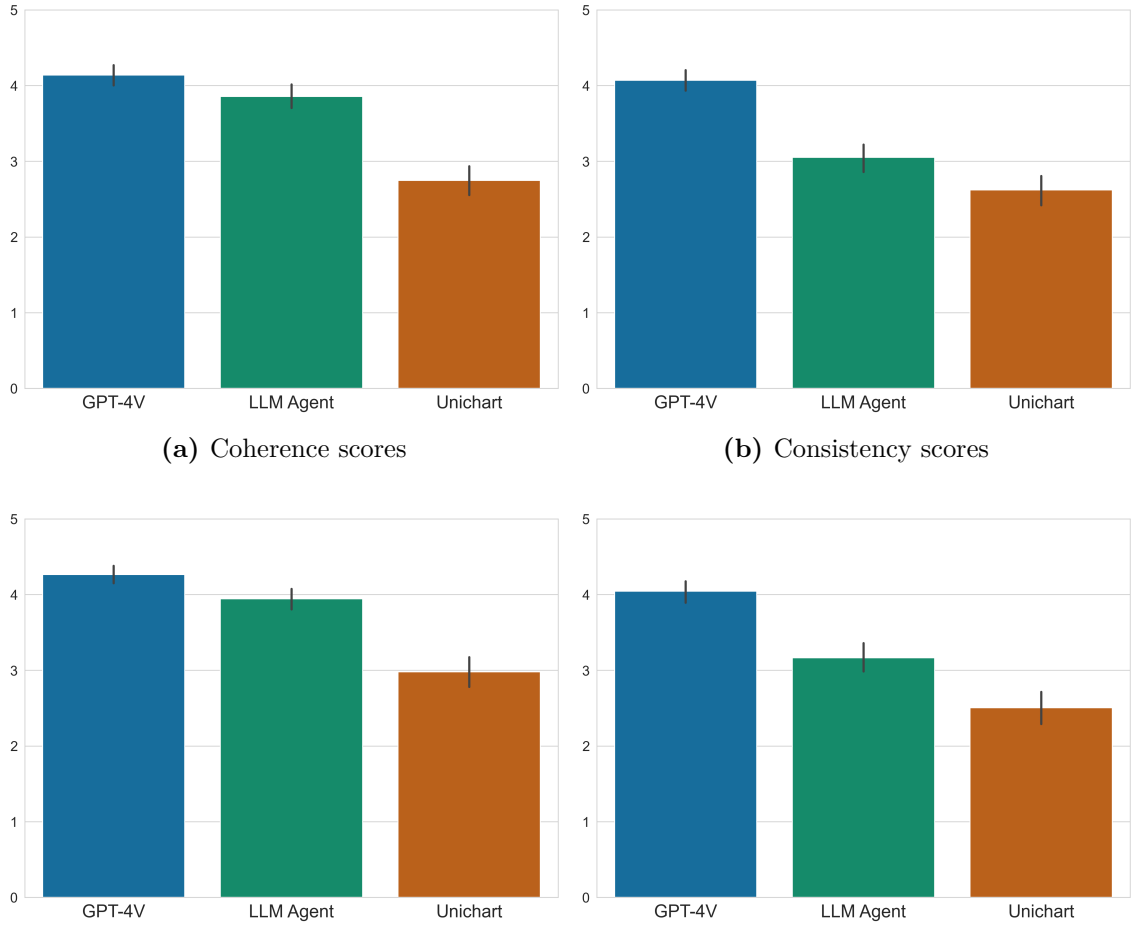
To understand the underlying reasons for these preferences, our next section examines the quality criteria, offering insights into each model's strengths and weaknesses for a more comprehensive evaluation of chart summarization performance.

4.5.2 Quality Criteria

Building upon our initial user preferences evaluation, this section presents a detailed analysis of model performance using the established quality criteria. As a brief reminder, the four key criteria are defined as follows: coherence measures the logical flow and organization of the summary; consistency evaluates the alignment between the summary and the original chart; fluency assesses the linguistic quality and readability of the summary; and relevance gauges how well the summary captures the essential information from the chart.

To compare the performance of GPT-4V, our LLM Agent, and Unichart across these criteria, we first conducted a descriptive analysis. Figure 4.5 presents the average scores and standard deviations for each model across all four criteria. The chart reveals that GPT-4V consistently achieved the highest average scores across all criteria, followed by our LLM Agent, with Unichart generally scoring the lowest. However, to determine the statistical significance of these differences, we employed a more rigorous statistical analysis.

Given our study design, which involved repeated measures (multiple evaluations by each participant) and ordinal data (Likert scale ratings), we chose the Friedman test as our initial statistical method. The Friedman test is particularly suitable for this scenario as it can



(a) Coherence scores

(b) Consistency scores

(c) Fluency scores

(d) Relevance scores

Figure 4.5: Comparison of Chart Summarization Model Performance Across Quality Criteria. This figure presents scores for GPT-4V, LLM Agent, and Unichart across four quality criteria: coherence, consistency, fluency, and relevance. GPT-4V consistently achieves the highest scores, followed by the LLM Agent, with Unichart generally scoring lowest. The performance gap is most pronounced in fluency and relevance. These results indicate GPT-4V's overall superiority, while highlighting the LLM Agent's competitive performance, particularly in linguistic aspects.

Criterion	Chi-square (χ^2)	p-value	Significant
Coherence	17.3947	10^{-4}	Yes
Consistency	14.8462	10^{-4}	Yes
Fluency	24.3377	10^{-6}	Yes
Relevance	16.3000	10^{-4}	Yes

Table 4.4: Friedman Test Results for Differences in Quality Scores Between Models. This table presents the chi-square values, p-values, and significance for four key quality criteria: coherence, consistency, fluency, and relevance. All criteria show statistically significant differences ($p < 0.001$) between the models

detect differences between three or more related groups (in our case, GPT-4V, LLM Agent, and Unichart) without assuming normality in the data distribution. Table 4.4 presents the results of the Friedman test for each quality criterion.

The Friedman test results indicate statistically significant differences ($p < 0.05$) among the models for all four quality criteria. This confirms that the choice of model indeed impacts the quality of chart summaries across all evaluated aspects. To identify specific pairwise differences between models, we conducted a Nemenyi post-hoc test. We selected this test due to its conservative nature in controlling for family-wise error rates, which is necessary when performing multiple comparisons. Table 4.5 summarizes the results of the Nemenyi post-hoc test, including p-values, mean rank differences, and standardized differences for each model pair across all criteria.

GPT-4V emerges as the leader in chart summarization, consistently outperforming Unichart across all criteria with statistically significant differences ($p < 0.05$) and large

Criterion	Model Pair	p-value	Significant	Mean Rank Diff.	Std. Diff.
Coherence	GPT-4V vs. LLM Agent	0.646000	No	0.27500	0.86963
	GPT-4V vs. Unichart	0.001000	Yes	1.22500	3.87379
	LLM Agent vs. Unichart	0.007508	Yes	0.95000	3.00416
Consistency	GPT-4V vs. LLM Agent	0.083060	No	0.67500	2.13454
	GPT-4V vs. Unichart	0.001000	Yes	1.20000	3.79473
	LLM Agent vs. Unichart	0.220962	No	0.52500	1.66020
Fluency	GPT-4V vs. LLM Agent	0.099311	No	0.65000	2.05548
	GPT-4V vs. Unichart	0.001000	Yes	1.52500	4.82247
	LLM Agent vs. Unichart	0.015617	Yes	0.87500	2.76699
Relevance	GPT-4V vs. LLM Agent	0.019684	Yes	0.85000	2.68794
	GPT-4V vs. Unichart	0.001000	Yes	1.25000	3.95285
	LLM Agent vs. Unichart	0.416890	No	0.40000	1.26491

Table 4.5: Nemenyi Post-hoc Test Results for Pairwise Comparisons of Chart Summarization Models. The table shows statistical significance (p-value < 0.05), mean rank differences, and standardized differences across four quality criteria, quantifying performance gaps between GPT-4V, LLM Agent, and Unichart.

standardized mean rank differences. The gap between GPT-4V and Unichart is particularly pronounced, with standardized differences ranging from 3.79473 to 4.82247 across the four criteria. This substantial performance difference suggests that GPT-4V's multimodal capabilities provide a significant advantage in interpreting and summarizing chart data, allowing it to excel in both linguistic and data-centric aspects of the task.

Our LLM Agent demonstrates a varied performance profile, showing strengths in certain areas while facing challenges in others. In language-related criteria, specifically coherence and fluency, the LLM Agent significantly outperforms Unichart ($p = 0.007508$ and $p = 0.015617$, respectively), with large standardized differences of 3.00416 and 2.76699. This

strong performance in linguistic aspects indicates that our strategy of leveraging GPT-4's language capabilities in the agent's design has been effective. Moreover, the differences between GPT-4V and our LLM Agent are not statistically significant for these criteria ($p = 0.646000$ for coherence and $p = 0.099311$ for fluency), suggesting that our agent achieves comparable linguistic quality to GPT-4V in generating summaries.

However, the LLM Agent faces challenges in data-centric aspects of chart summarization. In terms of consistency and relevance, our agent shows no significant difference from Unichart ($p = 0.220962$ and $p = 0.416890$, respectively), with relatively small standardized differences of 1.66020 and 1.26491. This similarity in performance is not unexpected, as our agent utilizes Unichart's chart derendering and CVQA capabilities for information extraction (see section 3.2.3.3 on tools used by the action module). The contrast becomes more apparent when comparing the LLM Agent to GPT-4V in these areas. GPT-4V significantly outperforms our agent in relevance ($p = 0.019684$, standardized difference = 2.68794) and shows a near-significant difference in consistency ($p = 0.083060$, standardized difference = 2.13454). These results highlight that data-centric aspects are areas where GPT-4V's native multimodal capabilities provide a distinct advantage.

While the Friedman and Nemenyi tests provided valuable insights into the overall performance differences between models, they do not account for individual participant variability or the relative importance of each criterion in determining model rankings. To address these limitations and gain a more complete understanding of how experts and

Group	Criterion	Correlation	95% CI	Mixed-Model Coef.	p-value
Expert	Coherence	-0.724	[-0.890, -0.388]	-0.402	1.01e-05
	Consistency	-0.687	[-0.874, -0.325]	-0.504	3.84e-14
	Fluency	-0.618	[-0.842, -0.213]	-0.458	3.63e-04
	Relevance	-0.665	[-0.864, -0.287]	-0.521	1.79e-26
Novice	Coherence	-0.116	[-0.848, 0.768]	-0.154	6.52e-01
	Consistency	-0.721	[-0.967, 0.219]	-1.032	9.77e-02
	Fluency	-0.319	[-0.898, 0.665]	-0.319	5.91e-01
	Relevance	-0.588	[-0.948, 0.427]	-2.697	3.62e-12

Table 4.6: Spearman Correlation and Mixed-Effects Model Analysis Comparing Expert and Novice Evaluations of Summarization Models. The table presents correlation coefficients, 95% confidence intervals, mixed-model coefficients, and p-values across the four quality criteria, quantifying the relationship between criteria scores and overall model rankings for expert and novice evaluators.

novices value chart summaries, we conducted an additional analysis using participant-level data.

This supplementary analysis employed two complementary statistical approaches: Spearman correlations with 95% confidence intervals and mixed-effects models. By using participant-level means for each model and criterion, we were able to preserve individual differences while reducing noise in the data. The Spearman correlations allow us to quantify the strength and direction of the relationship between each criterion and the overall model rankings, while the mixed-effects models account for both fixed effects (criteria) and random effects (participants). Table 4.6 presents the results of this analysis for both groups.

The results for the expert group demonstrate strong negative correlations between all

criteria and model rankings, with values ranging from -0.618 to -0.724. The relatively small range of these correlation values suggests that experts value all four criteria almost equally in appreciating chart summaries. This finding is further supported by the narrow confidence intervals and highly significant p-values (all $p < 0.001$) obtained from the mixed-effects models.

Interestingly, while coherence shows the strongest overall correlation (-0.724), the mixed-effects model, which accounts for individual participant differences, reveals that relevance and consistency have the largest coefficients (-0.5210 and -0.5041, respectively). This discrepancy suggests that when individual preferences are controlled for, data-centric criteria (relevance and consistency) emerge as more important in expert appreciations of chart summaries. However, the strong individual correlation for coherence indicates that some expert participants place high value on linguistic criteria.

The results for the novice group reveal a more complex picture. The correlations between criteria and rankings are generally weaker and not statistically significant, as evidenced by the wide confidence intervals that include zero. However, the mixed-effects model yields an intriguing result for relevance, showing a strong negative coefficient (-2.6970) with high statistical significance ($p = 3.62e - 12$).

This discrepancy between the correlation and mixed-effects model results for novices, particularly regarding relevance, warrants careful interpretation. The strong effect of relevance in the mixed-effects model suggests that when accounting for individual

differences, relevance may play a crucial role in novice evaluations of chart summaries. This finding could explain our previous observation: novices do not differentiate in their preference between Unichart and our LLM Agent, as these models have similar relevance scores (see Table 4.1). In contrast, experts, who appear to value linguistic properties of chart summaries more highly, do distinguish between Unichart (with lower linguistic quality scores) and our LLM Agent (with higher linguistic scores). This pattern suggests that while novice users might primarily value data-centric criteria, particularly relevance, expert users consider both data-centric and linguistic aspects in their appreciations. However, it is important to note that the small sample size for the novice group ($n = 2$) limits the reliability of these findings and necessitates further investigation with a larger cohort to confirm these patterns and the implied differences in evaluation priorities between novices and experts.

4.6 Discussion

This study aimed to evaluate the performance of our proposed LLM-based agent for chart summarization, compare it with existing baselines, and investigate the factors influencing user satisfaction across different expertise levels. Additionally, we sought to assess the effectiveness of our adapted quality criteria framework. The results of our user study and statistical analyses provide valuable insights into these research questions.

Addressing our first research question, the comparison between our LLM-based agent

and the baselines (GPT-4V and Unichart) revealed a diverse performance profile. GPT-4V consistently outperformed both our agent and Unichart across all quality criteria, demonstrating the superiority of VLLMs in chart summarization tasks. However, our LLM-based agent showed promising results, particularly in linguistic aspects of summarization. The agent significantly outperformed Unichart in coherence and fluency, with no statistically significant difference from GPT-4V in these criteria. Importantly, our agent achieved these results without relying on a dataset of labeled chart summaries, which was a primary goal of this thesis. In terms of overall preference, our model successfully outperformed Unichart, validating our approach's effectiveness. These findings suggest that our strategy of leveraging GPT-4's language capabilities in the agent's design effectively enhanced the linguistic quality of the generated summaries while maintaining independence from costly labeled datasets.

Nevertheless, our LLM-based agent faced challenges in data-centric aspects of summarization, specifically consistency and relevance. In these criteria, our agent's performance was comparable to Unichart but significantly lower than GPT-4V. This limitation in data extraction and interpretation is not surprising, considering that our system relies on Unichart's CVQA and chart derendering components for these tasks. Consequently, our proposed method offers a tangible advantage: improvements in chart summarization performance can be achieved by training more accurate and powerful CVQA and chart derendering models, without ever requiring costly labeled chart

summaries.

Regarding our second research question on factors influencing user satisfaction, we observed notable differences between expert and novice evaluators. Experts demonstrated a balanced appreciation for all four quality criteria, with strong negative correlations between each criterion and model rankings. This suggests that experts value a holistic approach to chart summarization, considering both linguistic and data-centric aspects almost equally. Interestingly, when accounting for individual differences through mixed-effects modeling, relevance and consistency emerged as slightly more influential in expert evaluations. This finding indicates that while all expert participants shared a common emphasis on data-centric criteria, some individuals valued linguistic criteria particularly highly.

In contrast, novice evaluators showed a distinct preference pattern. While correlation analyses for novices were inconclusive due to small sample size, the mixed-effects model revealed a strong influence of relevance on their appreciations of the summaries. This finding suggests that novice users prioritize the inclusion of key information from the chart over linguistic sophistication. These differences between expert and novice preferences highlight the need for further investigation into how various user groups value different aspects of chart summaries, which could inform the development of more targeted and effective summarization systems.

The effectiveness of our adapted quality criteria framework is evident in its ability to capture these fine differences in user preferences. By extending traditional text

summarization criteria to the visual-textual domain of chart summarization, we were able to provide a more comprehensive and informative assessment of summary quality. The framework's ability to distinguish between linguistic and data-centric aspects of summaries proved particularly valuable in identifying the strengths and weaknesses of different models and understanding user preferences across different user groups.

Based on these findings, we propose several design recommendations for chart summarization systems. First, developers should focus on creating systems that can adapt to the needs of different user groups. The observed differences between experts and novices highlight a spectrum of user preferences that may vary across other groups as well. Further investigation using the quality criteria framework could reveal additional insights into these diverse needs. Second, with the advent of powerful text-based large language models like LLaMA 3, achieving GPT-4V level performance in chart summarization may be possible by focusing efforts on developing stronger CVQA and chart derendering tools. These components are likely easier to train than open-source VLLMs at the level of GPT-4, and improvements in these areas would directly enhance the performance of modular systems like our LLM-based agent. Finally, the integration of diverse quality criteria in the development and evaluation processes can lead to more robust and user-centric summarization systems, allowing for targeted improvements and more informative performance assessments.

Chapter 5

Discussion

5.1 Summary of Themes and Key Findings

This thesis investigates the application of LLM-based agents for chart summarization, focusing on enhancing accessibility for visually impaired individuals and addressing the scarcity of labeled datasets in this domain. The research yields several findings that contribute to the field of AI-assisted data interpretation and accessibility technologies.

A key theme that emerged is the efficacy of the LLM-based agent approach in chart summarization. By leveraging the general capabilities of LLMs in conjunction with domain-specific tools, this method demonstrates the potential to advance the field without relying on extensive labeled datasets, addressing a significant challenge in chart summarization research.

The study also introduces the application of quality criteria from text summarization to chart summarization. This approach provides a framework for evaluation, offering insights into the assessment of chart summary quality and user preferences. The criteria, which include coherence, consistency, fluency, and relevance, allow for a multifaceted evaluation of generated summaries. This framework proved valuable in differentiating model performance across linguistic and data-centric aspects, as well as in identifying preference patterns among different user groups.

The empirical evaluation compared the performance of the LLM-based agent to two baselines: GPT-4V and Unichart. GPT-4V consistently outperformed both the LLM-based agent and Unichart across all quality criteria. However, the LLM-based agent demonstrated competitive performance in linguistic aspects, specifically coherence and fluency, where it matched GPT-4V and outperformed Unichart. In data-centric aspects such as consistency and relevance, the LLM-based agent performed similarly to Unichart but fell short of GPT-4V's capabilities.

Analysis of user preferences revealed differences between expert and novice evaluators. Experts demonstrated a balanced appreciation for all quality criteria, valuing both linguistic and data-centric aspects of summaries. In contrast, novices placed higher emphasis on relevance. A trend emerged suggesting that experts tended to prefer the LLM agent's summaries, while novices leaned towards Unichart's outputs, although this difference did not reach statistical significance.

5.2 Implications

The findings of this study have several implications for the field of chart summarization and AI-assisted data interpretation technologies.

The competitive performance of the LLM-based agent, particularly in linguistic aspects, supports the viability of this approach as a solution to dataset scarcity in chart summarization. This implies that combining general-purpose LLMs with specialized tools may offer a more resource-efficient path to advancing the state-of-the-art in this field. Rather than relying solely on extensive datasets of chart-summary pairs, future research efforts could focus on improving domain-specific tools for data extraction and interpretation.

However, it's important to note that the LLM-agent didn't surpass Unichart in data-centric aspects, indicating that the cognitive capabilities of LLMs didn't fully compensate for the limitations in data extraction and interpretation. This observation suggests a potential path forward for academia and open source to reach GPT-4 level (considering that models like Meta's LLAMA-3-70B are approaching this benchmark), while developing targeted, domain-specific models to enhance information extraction from charts. This strategy could involve improving existing tools like chart derendering and CVQA, as well as incorporating new tools such as OCR models to enhance data extraction capabilities by precisely capturing labels and values on charts.

The application of quality criteria from text summarization to chart summarization has

proven to be a valuable tool for advancing the field. These criteria enabled us to identify specific strengths and weaknesses in the models, providing clear direction for future development efforts. Moreover, they allowed us to discern differences in user preferences based on expertise levels. Our study demonstrates that these criteria can be used to compare model performance on highly relevant and interpretable dimensions. For this reason, we propose that these quality criteria should be adopted alongside user preference rankings as complementary approaches, serving as the golden standard in chart summarization research.

The observed differences in preferences between expert and novice users imply a need for customizable summary generation based on user expertise. This finding has potential applications in various settings. In educational environments, summarization systems could be tailored to provide more detailed explanations of chart elements and trends for novice students, while offering more concise, insight-focused summaries for advanced learners. The distinction made between experts and novices could be extended to other user categories, warranting further research into the preferences of different user types. This also highlights the need for summarization systems that are flexible enough to be customized for various user profiles. We believe that LLM-agents could be a step forward in achieving more flexible summarization, as their output can be modified through smart prompt engineering to cater to different user needs and preferences.

5.3 Limitations

While this study provides valuable insights into the potential of LLM-based agents for chart summarization, it is important to acknowledge several limitations that may impact the generalizability and interpretation of the results.

A primary limitation of this study is the sample size and diversity of participants, particularly in the user evaluation phase. The limited number of participants, especially in the novice category, may not fully represent the diverse range of potential users for chart summarization systems. Furthermore, the focus on charts from the economic domain, while providing a consistent context for evaluation, may limit the generalizability of the findings to other fields where chart summarization could be equally valuable, such as education or healthcare.

The current implementation of the LLM-based agent also presents certain limitations. While the agent demonstrated competitive performance in linguistic aspects of summarization, it only reached Unichart's level in data-centric aspects, showing notable gaps when compared to GPT-4V. Although prompt engineering was effective in acquiring information from tools and transforming it into summaries, it was not sufficient to overcome the inherent limitations of the original tools in information extraction.

The evaluation process, while informative, has its limitations. The quality criteria framework and preference rankings, though valuable, are time-intensive and costly, limiting scalability for larger studies. More critically, our study primarily measured user preference

rather than task-specific usefulness. The usefulness of chart summaries may vary significantly depending on the context and task at hand. For instance, an experiment measuring how effectively blind users can make stock trading decisions based on different summary types would provide a more direct measure of usefulness in a specific scenario. Such task-oriented evaluations could reveal that the most preferred summary style isn't always the most useful for complex decision-making tasks.

Reflections on the methodological choices reveal certain trade-offs. The modular approach of the LLM-based agent, while offering flexibility and the potential for incremental improvements through the addition of more advanced tools, introduces its own set of challenges. These include increased system complexity, the need for ongoing maintenance and updates of multiple components, and the intricacy of designing effective prompts for the agent to manage tool interactions. Notably, the processing time for the agent-based system designed was considerably longer compared to the other models, due to repeated sequential calls to the GPT-4 API. This increased processing time not only prevents the system's utilization in real-time scenarios but also hinders the overall user experience, potentially limiting its practical applicability.

An additional limitation is the static nature of the current summarization process. The system we designed doesn't allow for interactive discovery of the chart. In real-world applications, users might benefit from a more dynamic system where they could ask follow-up questions after the initial summarization to gain a deeper understanding of the

information at hand. The current approach, while valuable as a first step, lacks this interactive component that could enhance user engagement and comprehension.

5.4 Future Research Directions

Building upon the findings and limitations of this work, several promising avenues for future research emerge.

Methodological improvements represent an important area for future work. A priority should be conducting larger-scale studies with more diverse participant pools, particularly increasing the representation of novice users. This expanded research should also encompass participants from various fields beyond economics, such as education and healthcare, to test the generalizability of the findings. Furthermore, future studies should extend the range of chart types and domains analyzed, including those from scientific research, to evaluate the system's adaptability across different contexts.

To enhance the evaluation process, researchers should explore more efficient methods for applying the quality criteria framework and assessing user preferences, enabling larger-scale assessments. The use of LLMs as evaluators shows particular promise in both regards. For instance, adapting frameworks like G-EVAL, which uses LLMs to score text summaries on specific quality criteria through prompts that incorporate auto-generated chain-of-thought rationales and probability-weighted scoring, could facilitate the grading of quality criteria at scale. While G-EVAL has shown high correlations with human

judgments in text summarization, it would require careful adaptation to incorporate visual context for effective chart summary evaluation. Similarly, LLMs could be employed for preference ranking of chart summaries, providing a cost-effective means of approximating user preferences. However, it is important to validate these automated evaluation methods against human judgments to ensure their reliability in the context of chart summarization [78].

Another promising direction is the design and implementation of task-specific usefulness studies. For example, experiments measuring how effectively blind users can make stock trading decisions based on different summary types would provide direct insights into the practical value of chart summarization systems. Conducting such studies with the target user group – in this case, individuals with visual impairments – would yield potentially valuable results. These task-oriented evaluations could reveal important discrepancies between user preferences and actual usefulness in complex decision-making scenarios, potentially highlighting differences in how quality criteria manifest in practical applications.

On the technical front, improving the system's data-centric performance is a major area for improvement. Future research should focus on enhancing the capacity of the LLM-based agent to utilize tools more effectively. This could be achieved through the integration of VLLMs as specialized tools, leveraging their pre-training on visual data to improve chart interpretation. Additionally, improving the in-context learning of tools with more diverse

examples of tool usage and limitations could enhance the agent's adaptability. Implementing the tree of thoughts approach to replace chain of thought reasoning could lead to more robust and nuanced interpretations, potentially improving consistency and relevance scores [79]. To systematically develop and evaluate these enhancements, utilizing experiment tracking tools like ChainForge, coupled with our quality criteria framework and task-specific metrics, could lead to more robust and replicable improvements in the agent's performance [80].

Optimizing the modular approach presents opportunities for improvement. Researchers should investigate ways to enhance the system's processing time, potentially by utilizing smaller, distilled versions of strong open-source LLMs like LLAMA-3 [81]. While this could help reduce latency and improve the overall user experience, it's important to carefully balance model size reduction with maintaining high performance across all quality criteria, especially coherence and fluency where the current system excels.

Finally, future research should explore the development of interactive chart summarization systems. Creating a version that allows users to ask follow-up questions about the chart could enhance user engagement and comprehension. Comparative studies between static and interactive summarization systems would provide valuable insights into user preferences and the effectiveness of these approaches in different contexts, particularly in how interactivity might affect relevance and consistency scores.

5.5 Conclusion

This thesis explored the potential of LLM-based agents for chart summarization, with the primary objectives of developing an approach that addresses dataset scarcity and investigates ways to improve accessibility for visually impaired individuals. Through the development and evaluation of an LLM-based agent, we have made progress towards these goals and gained insights into the challenges and opportunities in this field.

Our research questions, focused on the viability of LLM-based agents for chart summarization and the impact of user expertise on summary preferences, have been addressed through empirical evaluation. The LLM-based agent demonstrated competitive performance in linguistic aspects, particularly in coherence and fluency, validating its potential as an alternative to traditional supervised learning approaches in these areas. However, its performance in data-centric aspects was comparable to existing methods, highlighting areas for future improvement. The analysis of user preferences revealed important distinctions between expert and novice users, emphasizing the need for adaptive summarization strategies that can cater to different levels of expertise.

The key contributions of this work include developing an LLM-based agent architecture in the context of chart summarization, which combines the general capabilities of large language models with specialized chart analysis tools, and adapting quality criteria from text summarization for the evaluation of chart summaries. These contributions provide insights into the challenges of applying LLM-based approaches to visual data interpretation

and offer a framework for evaluating chart summaries that could be useful in future research.

While our LLM-based agent did not outperform state-of-the-art models like GPT-4V, it demonstrated the potential of modular, tool-based approaches in addressing the challenge of dataset scarcity in chart summarization. This approach could be particularly valuable for academic and open-source research, where access to large proprietary models or extensive labeled datasets may be limited.

Looking ahead, there are several promising avenues for future research. These include improving the data extraction capabilities of the agent, exploring more efficient ways to leverage LLMs in the summarization process, and investigating methods for creating adaptive systems that can cater to different user needs and expertise levels. Additionally, further studies with larger and more diverse participant pools could provide more comprehensive insights into user preferences and the practical utility of different summarization approaches.

In conclusion, this thesis contributes to the ongoing research in AI-assisted data interpretation, specifically in the domain of chart summarization. While there is still much work to be done to create truly accessible and effective chart summarization systems, the insights gained from this study provide a stepping stone for future investigations in this area of research.

Bibliography

[1] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft coco: Common objects in context," 2015. [Online]. Available: https://arxiv.org/abs/1405.0312

[2] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik, "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models," 2016. [Online]. Available: https://arxiv.org/abs/1505.04870

[3] A. Masry, P. Kavehzadeh, X. L. Do, E. Hoque, and S. Joty, "UniChart: A Universal Vision-language Pretrained Model for Chart Comprehension and Reasoning," May 2023.

[4] S. Kantharaj, R. T. K. Leong, X. Lin, A. Masry, M. Thakkar, E. Hoque, and S. Joty, "Chart-to-Text: A Large-Scale Benchmark for Chart Summarization," Apr. 2022.

[5] R.-i. Taniguchi, M. Yokota, E. Kawaguchi, and T. Tamati, "Knowledge-based picture understanding of weather charts," *Pattern Recognition*, vol. 17, no. 1, pp. 109–123, Jan. 1984.

[6] W. Huang, C. L. Tan, and W. K. Leow, "Associating text and graphics for scientific chart understanding," in *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, Aug. 2005, pp. 580–584 Vol. 2.

[7] M. R. Morris, J. Johnson, C. L. Bennett, and E. Cutrell, "Rich Representations of Visual Content for Screen Reader Users," *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–11, Apr. 2018.

[8] R. A. Martínez, M. R. Turró, and T. G. Saltiveri, "Methodology for heuristic evaluation of the accessibility of statistical charts for people with low vision and color vision deficiency," Feb. 2021.

[9] A. Kumar, T. Ganu, and S. Guha, "ChartParser: Automatic Chart Parsing for Print-Impaired," Nov. 2022.

[10] P. Mishra, S. Kumar, M. K. Chaube, and U. Shrawankar, "ChartVi: Charts summarizer for visually impaired," *Journal of Computer Languages*, vol. 69, p. 101107, Apr. 2022.

[11] F. Liu, F. Piccinno, S. Krichene, C. Pang, K. Lee, M. Joshi, Y. Altun, N. Collier, and J. M. Eisenschlos, "MatCha: Enhancing Visual Language Pretraining with Math Reasoning and Chart Derendering," Dec. 2022.

[12] R. Wang, C. Jung, and Y. Kim, "Seeing Through Sounds: Mapping Auditory Dimensions to Data and Charts for People with Visual Impairments," *Computer Graphics Forum*, vol. 41, no. 3, pp. 71–83, 2022.

[13] Z. Zhang, "Tapsonic: One Dimensional Finger Mounted Multimodal Line Chart Reader," in *Proceedings of the 22nd International ACM SIGACCESS Conference on Computers and Accessibility*, ser. ASSETS '20. New York, NY, USA: Association for Computing Machinery, Oct. 2020, pp. 1–4.

[14] N. A. Grabowski and K. E. Barner, "Data visualization methods for the blind using force feedback and sonification," in *Telemanipulator and Telepresence Technologies V*, vol. 3524. SPIE, Dec. 1998, pp. 131–139.

[15] R. Isran, K. Sepehri, K. Theivendran, and A. Anwar, "Towards More Effective Data Visualization Methods Using Haptics," in *2021 IEEE World Haptics Conference (WHC)*. Montreal, QC, Canada: IEEE, Jul. 2021, pp. 590–590.

[16] J. Kim, A. Srinivasan, N. W. Kim, and Y.-S. Kim, "Exploring Chart Question Answering for Blind and Low Vision Users," in *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, ser. CHI '23. New York, NY, USA: Association for Computing Machinery, Apr. 2023, pp. 1–15.

[17] F. Bajić and J. Job, "Review of chart image detection and classification," *International Journal on Document Analysis and Recognition (IJDAR)*, Jan. 2023.

[18] H. Sharma and D. Padha, "A comprehensive survey on image captioning: From handcrafted to deep learning-based techniques, a taxonomy and open research issues," *Artificial Intelligence Review*, vol. 56, no. 11, pp. 13 619–13 661, Nov. 2023.

[19] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg, "BabyTalk: Understanding and Generating Simple Image Descriptions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2891–2903, Dec. 2013.

[20] V. Ordonez, G. Kulkarni, and T. Berg, "Im2Text: Describing Images Using 1 Million Captioned Photographs," in *Advances in Neural Information Processing Systems*, vol. 24. Curran Associates, Inc., 2011.

[21] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

[22] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.

[23] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and Tell: A Neural Image Caption Generator," Apr. 2015.

[24] H. Singh and S. Shekhar, "STL-CQA: Structure-based Transformers with Localization and Encoding for Chart Question Answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 3275–3284.

[25] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio, "Show, Attend and Tell: Neural Image Caption Generation with Visual Attention," Apr. 2016.

[26] A. Masry, D. X. Long, J. Q. Tan, S. Joty, and E. Hoque, "ChartQA: A Benchmark for Question Answering about Charts with Visual and Logical Reasoning," Mar. 2022.

[27] F. Liu, J. M. Eisenschlos, F. Piccinno, S. Krichene, C. Pang, K. Lee, M. Joshi, W. Chen, N. Collier, and Y. Altun, "DePlot: One-shot visual language reasoning by plot-to-table translation," Dec. 2022.

[28] S. O. ul Islam, I. Škrjanec, O. Dušek, and V. Demberg, "Tackling Hallucinations in Neural Chart Summarization," Aug. 2023.

[29] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," Mar. 2022.

[30] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei, "Scaling Instruction-Finetuned Language Models," Dec. 2022.

[31] K. Kafle, B. Price, S. Cohen, and C. Kanan, "DVQA: Understanding Data Visualizations via Question Answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5648–5656.

[32] S. E. Kahou, V. Michalski, A. Atkinson, A. Kadar, A. Trischler, and Y. Bengio, "FigureQA: An Annotated Figure Dataset for Visual Reasoning," Feb. 2018.

[33] R. Chaudhry, S. Shekhar, U. Gupta, P. Maneriker, P. Bansal, and A. Joshi, "LEAF-QA: Locate, Encode & Attend for Figure Question Answering," Jul. 2019.

[34] N. Methani, P. Ganguly, M. M. Khapra, and P. Kumar, "PlotQA: Reasoning over Scientific Plots," Feb. 2020.

[35] S. Kantharaj, X. L. Do, R. T. K. Leong, J. Q. Tan, E. Hoque, and S. Joty, "OpenCQA: Open-ended Question Answering with Charts," Oct. 2022.

[36] K. Wongsuphasawat, Z. Qu, D. Moritz, R. Chang, F. Ouk, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Voyager 2: Augmenting Visual Analysis with Partial View Specifications," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ser. CHI '17. New York, NY, USA: Association for Computing Machinery, May 2017, pp. 2648–2659.

[37] Z. Xi, W. Chen, X. Guo, W. He, Y. Ding, B. Hong, M. Zhang, J. Wang, S. Jin, E. Zhou, R. Zheng, X. Fan, X. Wang, L. Xiong, Y. Zhou, W. Wang, C. Jiang, Y. Zou, X. Liu, Z. Yin, S. Dou, R. Weng, W. Cheng, Q. Zhang, W. Qin, Y. Zheng, X. Qiu, X. Huang, and T. Gui, "The Rise and Potential of Large Language Model Based Agents: A Survey," Sep. 2023.

[38] A. Newell and H. A. Simon, "Computer science as empirical inquiry: Symbols and search," in *ACM Turing Award Lectures*. New York, NY, USA: Association for Computing Machinery, 1975.

[39] M. Ginsberg, *Essentials of Artificial Intelligence*. Elsevier, 1993.

[40] R. A. Brooks, "Intelligence without representation," *Artificial Intelligence*, vol. 47, no. 1, pp. 139–159, Jan. 1991.

[41] L. P. Kaelbling, "An Architecture For Intelligent Reactive Systems," in *Reasoning About Actions & Plans*. Elsevier, 1987, pp. 395–410.

[42] C. Ribeiro, "Reinforcement Learning Agents," *Artificial Intelligence Review*, vol. 17, no. 3, pp. 223–250, May 2002.

[43] Y. Li, "Deep Reinforcement Learning: An Overview," Nov. 2018.

[44] E. Parisotto, J. L. Ba, and R. Salakhutdinov, "Actor-Mimic: Deep Multitask and Transfer Reinforcement Learning," Feb. 2016.

[45] R. Fakoor, P. Chaudhari, S. Soatto, and A. J. Smola, "Meta-Q-Learning," Apr. 2020.

[46] OpenAI and Others, "GPT-4 Technical Report," Mar. 2024, full author list includes 280 contributors.

[47] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," Jan. 2023.

[48] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large Language Models are Zero-Shot Reasoners," Jan. 2023.

[49] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou, "Self-Consistency Improves Chain of Thought Reasoning in Language Models," Mar. 2023.

[50] B. Peng, M. Galley, P. He, H. Cheng, Y. Xie, Y. Hu, Q. Huang, L. Liden, Z. Yu, W. Chen, and J. Gao, "Check Your Facts and Try Again: Improving Large Language Models with External Knowledge and Automated Feedback," Mar. 2023.

[51] C. Qian, X. Cong, W. Liu, C. Yang, W. Chen, Y. Su, Y. Dang, J. Li, J. Xu, D. Li, Z. Liu, and M. Sun, "Communicative Agents for Software Development," Dec. 2023.

[52] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language Models are Few-Shot Learners," Jul. 2020.

[53] A. Lu, H. Zhang, Y. Zhang, X. Wang, and D. Yang, "Bounding the Capabilities of Large Language Models in Open Text Generation with Prompt Constraints," Feb. 2023.

[54] N. McKenna, T. Li, L. Cheng, M. J. Hosseini, M. Johnson, and M. Steedman, "Sources of Hallucination by Large Language Models on Inference Tasks," Oct. 2023.

[55] H. H. Park, Y. Vyas, and K. Shah, "Efficient Classification of Long Documents Using Transformers," Mar. 2022.

[56] N. Shinn, F. Cassano, E. Berman, A. Gopinath, K. Narasimhan, and S. Yao, "Reflexion: Language Agents with Verbal Reinforcement Learning," Oct. 2023.

[57] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," Jun. 2021.

[58] OpenAI, "Gpt-4 vision system card," Sep. 2023. [Online]. Available: https://cdn.openai.com/papers/GPTV_System_Card.pdf

[59] J. Li, D. Li, S. Savarese, and S. Hoi, "BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models," Jan. 2023.

[60] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, "ReAct: Synergizing Reasoning and Acting in Language Models," Mar. 2023.

[61] J. Huang and K. C.-C. Chang, "Towards Reasoning in Large Language Models: A Survey," May 2023.

[62] G. Kim, P. Baldi, and S. McAleer, "Language Models can Solve Computer Tasks," Nov. 2023.

[63] Y. Zhang, J. Henkel, A. Floratou, J. Cahoon, S. Deep, and J. M. Patel, "ReAcTable: Enhancing ReAct for Table Question Answering," Oct. 2023.

[64] Y. Qin, S. Liang, Y. Ye, K. Zhu, L. Yan, Y. Lu, Y. Lin, X. Cong, X. Tang, B. Qian, S. Zhao, R. Tian, R. Xie, J. Zhou, M. Gerstein, D. Li, Z. Liu, and M. Sun, "ToolLLM: Facilitating Large Language Models to Master 16000+ Real-world APIs," Jul. 2023.

[65] S. Liu, L. Fan, E. Johns, Z. Yu, C. Xiao, and A. Anandkumar, "Prismer: A Vision-Language Model with An Ensemble of Experts," Mar. 2023.

[66] C. Wu, S. Yin, W. Qi, X. Wang, Z. Tang, and N. Duan, "Visual ChatGPT: Talking, Drawing and Editing with Visual Foundation Models," Mar. 2023.

[67] W. Kryscinski, N. S. Keskar, B. McCann, C. Xiong, and R. Socher, "Neural Text Summarization: A Critical Evaluation," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 540–551.

[68] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating Text Generation with BERT," Feb. 2020.

[69] A. R. Fabbri, W. Kryściński, B. McCann, C. Xiong, R. Socher, and D. Radev, "SummEval: Re-evaluating Summarization Evaluation," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 391–409, Apr. 2021.

[70] D. Yadav, J. Desai, and A. K. Yadav, "Automatic Text Summarization Methods: A Comprehensive Review," Mar. 2022.

[71] OpenAI, "OpenAI Models Documentation." [Online]. Available: https://platform.openai.com/docs/models

[72] F. Yan, H. Mao, C. C.-J. Ji, T. Zhang, S. G. Patil, I. Stoica, and J. E. Gonzalez, "Berkeley function calling leaderboard," https://gorilla.cs.berkeley.edu/blogs/8_berkeley_function_calling_leaderboard.html, 2024.

[73] W. Kintsch and T. A. van Dijk, "Toward a model of text comprehension and production," *Psychological Review*, vol. 85, no. 5, pp. 363–394, 1978.

[74] G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information," *Psychological Review*, vol. 63, no. 2, pp. 81–97, 1956.

[75] H. MacLeod, C. L. Bennett, M. R. Morris, and E. Cutrell, "Understanding Blind People's Experiences with Computer-Generated Captions of Social Media Images," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ser. CHI '17. New York, NY, USA: Association for Computing Machinery, May 2017, pp. 5988–5999.

[76] O. Ernst, O. Shapira, I. Dagan, and R. Levy, "Re-Examining Summarization Evaluation across Multiple Quality Criteria," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 13 829–13 838.

[77] D. Gillick and Y. Liu, "Non-Expert Evaluation of Summarization Systems is Risky," in *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, C. Callison-Burch and M. Dredze, Eds. Los Angeles: Association for Computational Linguistics, Jun. 2010, pp. 148–151.

[78] Y. Liu, D. Iter, Y. Xu, S. Wang, R. Xu, and C. Zhu, "G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment," May 2023.

[79] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan, "Tree of Thoughts: Deliberate Problem Solving with Large Language Models," May 2023.

[80] I. Arawjo, C. Swoopes, P. Vaithilingam, M. Wattenberg, and E. Glassman, "Chainforge: A visual toolkit for prompt engineering and llm hypothesis testing," 2023.

[81] A. Dubey and Others, "The llama 3 herd of models," 2024, full author list includes 280 contributors. [Online]. Available: https://arxiv.org/abs/2407.21783

Appendix A

Expertise Questionnaire

1. Which of the following best describes the relationship between inflation and unemployment rates in the short run? (Medium)

a) No relationship

b) Direct, positive relationship

c) Inverse relationship, as described by the Phillips Curve

d) Always move in the same direction

2. How does the Consumer Price Index (CPI) measure inflation? (Easy)

a) Tracking price changes in a fixed basket of goods/services

b) Measuring quarterly changes in GDP

c) Calculating average price of a changing basket of goods/services

d) Surveying consumers on perceived living costs

3. Which of the following is a potential cause of cost-push inflation? (Medium)

a) Increasing consumer demand

b) Rising wages due to labor shortages

c) Expansionary monetary policy

d) Increasing global competition

4. Which of the following is NOT a component of GDP? (Easy)

a) Consumer spending

b) Government spending

c) Stock market performance

d) Net exports

5. Central banks can indirectly influence which of the following? (Hard)

a) Government tax rates

b) Bank lending rates

c) Corporate hiring decisions

d) Stock market valuations

6. Which of the following is an example of a progressive tax system? (Medium)

a) A flat income tax rate for all taxpayers

b) Higher income tax rates for lower-income earners

c) Lower sales tax rates on luxury goods

d) Higher income tax rates for higher-income earners

7. The unemployment rate is calculated as: (Easy)

a) # unemployed / total population

b) # unemployed / labor force

c) # employed / labor force

d) # unemployed / # employed

8. Which of the following would be considered a discouraged worker? (Hard)

a) A person who is actively seeking work but unable to find a job that matches their qualifications

b) A person who wants a job but has given up looking due to lack of opportunities

c) A person who is working part-time but wants to work full-time for better pay and benefits

d) A person who is unemployed and not actively seeking work due to family responsibilities

9. A bear market refers to: (Easy)

a) A period of rising stock prices

b) A period of falling stock prices

c) High volatility in the housing market

d) Rapid growth in the money supply

10. Which of the following best describes the primary goal of monetary policy? (Medium)

a) Achieving price stability and sustainable economic growth

b) Balancing the government budget

c) Increasing government spending on infrastructure

d) Encouraging foreign investment in domestic markets

11. Which of the following is generally considered a leading indicator for the housing market? (Medium)

a) New home sales

b) Existing home sales

c) Housing starts

d) Home price appreciation

12. Which of the following is most likely to contribute to increasing wealth inequality? (Hard)

a) Regressive taxation policies that benefit high-income earners

b) Increasing prevalence of part-time and contract work arrangements

c) Stock market gains accruing primarily to the wealthy

d) Uneven access to healthcare and health insurance based on socioeconomic status

13. The labor force participation rate measures: (Medium)

a) The percentage of the total population that is employed

b) The percentage of the working-age population that is either employed or actively seeking work

c) The percentage of the working-age population that is employed

d) The percentage of the total population that is unemployed

14. Which of the following best describes the impact of rising interest rates on the housing market? (Hard)

a) Encourages more people to buy homes before interest rates increase further

b) Decreases affordability and demand for homes

c) Leads to a higher supply of homes as builders rush to complete projects

d) Increases the number of adjustable-rate mortgages as buyers seek lower initial payments

15. Which of the following is an example of expansionary fiscal policy? (Medium)

a) Raising taxes

b) Cutting government spending

c) Increasing government spending on infrastructure projects

d) The central bank selling government bonds