From Human-Graph to Agent-Graph Interaction: State of the Art and Future Directions

Sourav S Bhowmick
Nanyang Technological University
Singapore
assourav@ntu.edu.sg

Arijit Khan
Bowling Green State University
United States
arijitk@bgsu.edu

Byron Choi
Hong Kong Baptist University
Hong Kong SAR, China
choi@hkbu.edu.hk

ABSTRACT

The multidisciplinary field of *Human-Data Interaction (HDI)* investigates human engagement with data, conceptually extending the paradigm of Human-Computer Interaction (HCI) beyond computer interfaces to data systems. As a specialized subfield within HDI, *Human-Graph Interaction (HGI)* focuses on the cognitive and procedural mechanisms through which individuals interpret, navigate, and analyze graph-structured representations. Concurrently, developments in artificial intelligence (AI) have facilitated *Agent-Graph Interaction (AGI)*, a domain in which autonomous agents reason over, manipulate, and execute operations on graph abstractions to perform user-defined tasks. This tutorial offers the first *unified* survey of HGI and AGI, reviewing methods that support human interaction and agentic capabilities such as planning, execution, and coordination. We highlight shared and complimentary challenges, and research opportunities for the data management community.

PVLDB Reference Format:
Sourav S Bhowmick, Arijit Khan, and Byron Choi. From Human-Graph to Agent-Graph Interaction: State of the Art and Future Directions. PVLDB, 19(12): 4878 - 4883, 2026.
doi:10.14778/3827998.3828143

1 INTRODUCTION

Human-Graph Interaction (HGI) studies how users understand, explore, and analyze graph-structured data. As an instance of *Human-Data Interaction (HDI)* [57], it integrates insights from multiple disciplines (*e.g.,* computer science, cognitive psychology, human-computer interaction (HCI)) to design effective tools and systems for supporting graph-centered interactions (see Section 2.2 for details). Similar to HDI, HGI focuses on visualization, navigation, and querying, addressing challenges such as complexity, scalability, and cognitive load in extracting meaningful insights from graphs.

While humans have directly interacted with data for decades, rapid advances in large language models (LLMs) and AI agents [59] in recent years have made it feasible for humans to delegate an expanding range of tasks to autonomous systems capable of interpreting goals, understanding context, and acting without continuous human oversight. This shift has given rise to the paradigm of *Agent-Data Interaction (ADI)*, in which autonomous AI agents access, interpret, transform, and leverage data to make decisions or complete tasks. Within this emerging paradigm, there is growing interest in *Agent-Graph Interaction (AGI)*[1] where agents model information as graphs (*e.g.,* knowledge graph) [4] and interact with these structures to carry out user-specified objectives.

Building upon these recent developments, this tutorial provides a comprehensive review of the emerging literature on both HGI and AGI, systematically examining how each paradigm addresses the challenges of **interacting** with graph-structured data and representations. Regarding HGI, we survey foundational methodologies and systems designed to facilitate human interaction with graph schemas, queries, and underlying data. Within AGI, we examine emerging techniques that empower autonomous agents to model, navigate, and execute operations on diverse graph abstractions. By systematically synthesizing these two paradigms, this tutorial highlights the conceptual convergence between human and agent operational modalities on graphs, emphasizing their shared computational challenges, complementary capabilities, and prospective avenues for future research at their intersection.

The scope of this tutorial is summarized below. While several surveys and tutorials cover graph-related topics (*e.g.,* [4, 6, 9, 33, 41]), to the best of our knowledge, none systematically address this combined focus.

- **Unpacking HGI and AGI:** We first introduce HDI and position HGI as a specialized domain within it. Next, we examine AGI under the broader umbrella of Agent-Data Interaction (ADI), highlighting its shared challenges and distinct characteristics relative to HGI.
- **Facets of human interactions with graphs:** In this part, we review various frameworks presented in the literature for interacting with graph schemas, queries, and data and their contributions to the advancement of HGI.
- **Facets of agentic interactions with graphs:** In this part, we review techniques from the AI literature that enable agentic interaction with diverse graph abstractions, supporting four core agent capabilities: *task planning*, *task execution*, *memory management*, and *multi-agent coordination*. Crucially, we highlight the conceptual convergence between these agentic capabilities and human cognitive modalities in HGI – specifically drawing parallels between agent decision-making processes and human engagement with graph schemas, queries, and data.
- **Future research directions:** Finally, we outline key open problems in human-graph and agent-graph interactions and discuss the data management community's potential role.

Proceedings of the VLDB Endowment, Vol. 19, No. 12 ISSN 2150-8097.
doi:10.14778/3827998.3828143

[1] In this paper, *AGI* refers strictly to *Agent-Graph Interaction*, rather than *Artificial General Intelligence*.

Table 1: Overview of topics.

Topic	*Key Papers*	*Demo*
Introduction		
Overview of HDI, HGI, and AGI	[4, 11, 20, 28, 57]	No
Human-Graph Interaction (HGI)		
Human Interaction with Graph Schemas	[1, 13]	Yes ([14])
Human Interaction with Graph Queries	[7, 8, 33, 47, 62]	Yes ([48])
Human Interaction with Graph Data	[7, 24, 51, 58, 71]	Yes ([58])
Agent-Graph Interaction (AGI)		
Agent Interaction with Task Planning Graph	[21, 36, 56, 66, 74]	No
Agent Interaction with Task Execution Graph	[12, 42, 43]	No
Agent Interaction with Memory Graph	[27, 29, 67, 68]	No
Multi-agent Interaction with Agent Coordination Graph	[4, 26]	No
Future Research Directions		

2 TUTORIAL OUTLINE

Organized top-down across four sections, this tutorial first establishes its scope by defining *Human-Data Interaction (HDI)*, *Human-Graph Interaction (HGI)*, and *Agent-Graph Interaction (AGI)*. It then details HGI methods for interacting with graph schemas, queries, and data representations, followed by an exploration of LLM-agent engagement with graph-structured abstractions in AGI. The presentation concludes with open research directions. Table 1 outlines the organization of the tutorial: the key papers covered and live tool demonstrations.

2.1 OVERVIEW: HDI, HGI, AND AGI

Human–Data Interaction (HDI) studies how people engage with large, complex, and heterogeneous datasets. It integrates perspectives from computer science, human–computer interaction (HCI), cognitive psychology, and design to build systems that enable effective data understanding and decision-making [57]. Although closely related, HDI and HCI differ in scope. HCI broadly focuses on the design and evaluation of interactive computing systems, whereas HDI specifically examines how users access, interpret, and control data within those systems. Thus, HCI is system-oriented, while HDI is data-oriented.

Definition of HDI. Definitions of HDI vary. Some emphasize human sensemaking through embodied, physical interactions with data. For example, Elmqvist [20] uses *"HDI"* to refer to the *"human manipulation, analysis, and sensemaking of large, unstructured, and complex datasets."* Others focus on embodied interaction techniques that facilitate exploration. For instance, Cafaro [11] observed that HDI encompasses *"technologies that use embodied interaction to facilitate the users' exploration of rich datasets."* A broader view conceptualizes HDI as the interplay between humans, data, and algorithms, including how human decisions generate data and how algorithmic outputs influence users [57].

Hornung *et al.* [28] adopt a human-centered perspective on HDI, emphasizing that one of its main goals should be designing interactions that help stakeholders achieve desired outcomes while avoiding unintended consequences. They argue that understanding and shaping the consequences of data use requires considering complex contextual factors, including the beliefs, values, and norms of those involved.

Evaluating HDI Frameworks. To evaluate HDI systems, prior work [57] proposes several criteria, including clarity, intuitiveness, informativeness, usability, and aesthetic qualities such as elegance and attractiveness. Effective systems should also provide intuitive feedback and support semantically meaningful data exploration.

Human–Graph Interaction (HGI). HGI is a specific form of HDI centered around graph data, which is widely used to model relationships in domains such as social networks, biology, and knowledge representation [53]. We focus on frameworks and techniques that facilitate embodied interaction, thereby assisting diverse users in exploring and making sense of graph datasets. Specifically, we examine user interaction across three primary facets of graph data systems: (i) *schemas*, which define structure and constraints, (ii) *queries*, which specify patterns of interest, and (iii) *data*, including raw graphs and query results. We elaborate on them in Section 2.2.

Agent–Graph Interaction (AGI). Recent advances in LLMs have enabled a new paradigm: *Agent–Graph Interaction (AGI)*, where autonomous agents, rather than humans, interact with graph-structured data to perform tasks. An *AI agent* perceives its environment, makes decisions, and executes actions to achieve goals. LLM-based agents extend this capability with language understanding, planning, memory, and tool usage [64]. Graphs provide a natural representation for many aspects of agent behavior, including reasoning, dependencies, and memory structures. Consequently, AGI explores how agents leverage graphs for *task planning*, *execution*, *memory management*, and *coordination* [4]. We elaborate on these facets of interactions in Section 2.3.

Conceptual Convergence of HGI and AGI. Although human interactions involving schemas, queries, and graph data superficially diverge from agentic operations over graph abstractions during execution and decision-making, Table 2 delineates their conceptual parallels, thereby establishing a bridge between HGI and AGI.

2.2 HUMAN–GRAPH INTERACTION (HGI)

To the best of our knowledge, HGI has not yet been given a widely accepted definition. Accordingly, we build upon the different definitions of HDI introduced in Section 2.1 and define HGI as follows: *"human manipulation, analysis, exploration, and sensemaking of graph-structured datasets to achieve desired outcomes."* In this section, we review the frameworks that facilitate it.

HGI generally adopts either a *visual* or a *textual* modality [6, 7]. We prioritize visual interaction frameworks in this tutorial due to distinct cognitive and practical advantages. Cognitively, the visual cortex leverages parallel, preattentive processing to interpret data, avoiding the high working-memory demands of sequential textual decoding [17, 63]. Practically, visual interfaces lower the barrier to entry by abstracting formal query and schema syntax, rendering complex graph analysis intuitive to domain specialists [6]. Consequently, this approach adheres to fundamental HDI imperatives, including usability, intuitiveness, and aesthetic engagement.

2.2.1 Interaction with Graph Schemas. Graph schemas organize data and enforce constraints, serving both descriptive (guiding users or systems) and prescriptive (enforcing data constraints) roles. However, practical support for schema interaction remains limited [52].

The LSG Abstraction. A popular abstraction is the *labeled schema graph* (LSG) [1], where node and edge types are represented using labeled shapes and connections. While simple, LSG suffers from visual clutter due to textual labels and limited expressiveness for advanced schema features such as constraints and label expressions.

The LICS Abstraction. To address the limitations of LSG, the *labeled iconized composite schema* (LICS) abstraction has been proposed in

Table 2: Conceptual bridge between HGI and AGI.

HGI Facet	*AGI Facet*	*Conceptual Bridge*
Graph Schema Interaction	Interaction with Task Planning Graph	Schemas serve as "planning" abstractions for humans, facilitating the understanding, retrieval, and exploration of graph data, whereas the Task Planning Graph serves as a planning "schema" that enables AI agents to execute a given task.
Graph Query Interaction	Interaction with Task Execution Graph	Graph queries enable humans to navigate graph data through query execution whereas Task Execution Graphs enable agents to navigate relevant data through task execution.
Graph Data Interaction	Interaction with Memory Graph	Graph data interactions enable humans to explore and retrieve information, whereas memory graph interactions enable agents to explore and retrieve contextual knowledge.

the context of property graphs [13]. LICS uses composite geometric *shapes* and *icons* to represent schema elements effectively, guided by principles and theories from HCI, visualization, and cognitive psychology. This reduces visual clutter while improving expressiveness. It is implemented using a *Map–Paint* paradigm, where schema elements are first mapped to visual categories and then rendered by leveraging various relevant theories and principles. Tools such as PASCAL [14] demonstrate how LICS supports intuitive visualization and exploration of moderate-complexity schemas.

2.2.2 Interaction with Graph Queries. Query formulation and execution constitute a primary paradigm for user interaction with graph data systems.

The LAG Abstraction. Early approaches rely on the *labeled atomic graph* (LAG) abstraction [7, 62], where nodes and edges are represented as simple shapes and queries are constructed through direct manipulation. Common query construction methods include: (i) edge-at-a-time, (ii) pattern-at-a-time, and (iii) query-by-example. Most LAG-based systems use manual interface design, which limits flexibility and adaptability. Data-driven approaches improve this by dynamically generating interfaces based on underlying data [8]. However, LAG struggles with complex graph models such as property and knowledge graphs, where multiple predicates per node or edge lead to cluttered or incomplete representations.

The LCG Abstraction. The *labeled composite graph* (LCG) abstraction addresses these issues in the context of property graphs by using composite visual elements to encode predicates [47]. Node and edge properties are represented through color, size, and structure, enabling richer visualization without additional user interaction. The LCG offers not only richer features than the LAG model, but also its design embraces principles from HCI, visualization, and cognitive psychology to enhance usability and aesthetics. LCG is supported by the VEDA [47] visual language and implemented in systems such as SIERRA [48], which facilitate intuitive property graph query construction and visualization.

2.2.3 Interaction with Graph Data. Finally, we review efforts to visually interact with graph data and query results.

Direct Interaction With Graphs. Direct interaction with large graphs is challenging due to their topological complexity [70]. To address this, visualization techniques such as hybrid layouts and graph thumbnails improve readability and navigation [3, 24, 51, 71].

Interaction Through Query Results. Another approach focuses on interaction through query results. For a set of small– or medium–sized graphs (*e.g.,* chemical compounds), systems highlight matching subgraphs and support ranking based on similarity. Systems like FERRARI [58] enable iterative query formulation and exploration, where users refine queries based on intermediate results. For large networks (*e.g.,* social networks), three main strategies are used: (i) summarization-based methods, (ii) region-based exploration, and (iii) feature-based techniques [7].

Interaction Through Summarization. Graph summarization offers an alternative by compressing entire datasets into manageable representations, preserving essential structure while reducing complexity. A variety of graph summarization techniques have been proposed in the literature (*e.g.,* [41]). While these efforts primarily focus on algorithmic techniques for summary construction, the systematic investigation of human interaction with such summaries remains largely underexplored.

2.3 AGENT-GRAPH INTERACTION (AGI)

In the previous section, we surveyed research on human-graph interaction, focusing on how people engage with graphs to carry out interactive tasks. With the rapid advancement of large language models (LLMs) and AI agents, however, it has become increasingly feasible for humans to delegate various tasks to autonomous systems. Accordingly, we now turn to recent work on AI agents interacting with graph-structured data/abstraction to perform user-specified tasks on behalf of humans. We define AGI as follows: *"Manipulation, analysis, exploration, and sensemaking of graphs by AI agents to achieve human-defined tasks."*

2.3.1 Agent Interaction With Task Planning Graphs. Planning is a fundamental capability of AI agents, requiring them to interpret user requests and generate structured sequences of actions [30]. Graph-based representations play a central role by supporting reasoning, task decomposition, and decision-making.

Knowledge Graph for Task Reasoning. First, knowledge graphs (KGs) improve task reasoning by providing rich, multi-hop contextual information. Agents interact with KGs to retrieve relevant subgraphs based on task-specific entities and relationships, enabling deeper understanding and more informed reasoning. Methods such as ToG [55], KG-CoT [74], RoG [45], MindMap [61], and PoG [56] use KG retrieval, relation-path generation, and multi-stage graph exploration to enhance this capability, while some approaches also represent an agent's internal reasoning as graph structures (*e.g.,* GoT [69] and *Graph of Thoughts* [5]).

Task Dependency Graph for Task Decomposition. Task decomposition can be modeled using *task dependency graphs (TDGs)*, which represent subtasks as nodes and their dependencies as directed edges in an acyclic structure. Agents use TDGs to determine execution order and identify feasible subtask sequences. Existing approaches to TDG-based planning include LLM-based methods that rely on textual representations (*e.g.,* AgentKit [65], Villager-Agent [18]), reinforcement learning approaches that optimize execution paths through reward mechanisms (*e.g.,* DAG-Plan [21], LGC-MARL [31]), and graph neural network methods that leverage structural information to select optimal subtask sequences (*e.g.,* FGRL [49], GNN4TaskPlan [66]).

State Space Graph for Decision Search. Decision-making during planning is supported by *state space graphs (SSGs)*, where nodes represent possible states and edges represent transitions between them. Agents perform search over these graphs to identify action sequences that lead to desired goals (*e.g.,* GBOP [36]).

Together, these graph-based frameworks provide a unified foundation for improving the efficiency, scalability, and effectiveness of AI agent planning.

2.3.2 Agent Interaction With Task Execution Graphs. The execution phase of an AI agent involves carrying out planned actions by leveraging both external tools and interactions with its environment. Graph-based representations play a crucial role in organizing these processes and enabling efficient execution.

Tool Graph for Tool Usage. Tool usage can be structured through *tool graphs*, which explicitly represent tools (*e.g.,* APIs, functions, or libraries) as nodes and their dependencies or information flow as edges. By interacting with such graphs, agents can determine appropriate sequences of tool invocations for a given task. Approaches model tool usage as directed acyclic graphs [75], restrict possible tool selections based on sparse usage patterns [42, 44], or construct unified graphs of tools and resources [43]. These strategies improve efficiency, scalability, and reliability, while also reducing errors such as hallucinated tool calls.

Environment-Interaction Graph. Environment interaction is facilitated through *environment-interaction graphs*, which model relationships between agents and environmental entities, including spatial, functional, or dependency-based connections. These graphs can be constructed using heuristic rules or learned dynamically from data. Examples include *scene graphs* [12] that provide structured representations of environments and learning-based methods that adaptively update relationships using attention mechanisms [22, 50]. By interacting with them, agents can better perceive relevant environmental factors and make informed decisions.

Overall, graph–based abstractions enable agents to coordinate tool usage and environment interaction effectively, enhancing execution accuracy, adaptability, and performance in complex tasks.

2.3.3 Agent Interaction With Memory Graph. Memory enables AI agents to store and recall experiences, improving accuracy and adaptability over time. Graph-based memory structures are particularly effective, as they capture relationships among information, enhancing pattern recognition and long-term reasoning. Knowledge graphs are used for long-term memory [2], often organized hierarchically to support both fine-grained details and high-level abstractions [15, 19, 68].

Graphs for Memory Retrieval. For memory retrieval, graph-based retrieval-augmented generation (RAG) techniques allow agents to efficiently access relevant information. Some methods improve accuracy by identifying meaningful subgraphs using semantic and structural signals [25, 27, 40, 46], while others enhance efficiency through optimized subgraph selection and path pruning, reducing computational cost and latency [16, 29].

Graphs for Memory Maintenance. Memory maintenance focuses on dynamically updating graph structures as new information becomes available. Recent approaches introduce adaptive graph architectures that incrementally incorporate new experiences while preserving consistency [67]. Hierarchical dynamic graphs further support tracking both immediate context and long-term goals [68]. Efficient update mechanisms such as incremental expansion [23, 25], language model–guided updates [32], and reinforcement learning strategies [60] help maintain scalability and relevance.

Overall, graph–based memory systems enable agents to retrieve and update knowledge effectively, supporting robust long-term learning and reasoning.

2.3.4 Graph for Multi-agent Interaction. In multi-agent systems, graphs model communication and coordination [26]. An *agent coordination graph (ACG)* represents agents as nodes and communications as edges [4]. These graphs can be dynamically learned to optimize coordination strategies and improve task performance.

2.4 THE ROAD AHEAD

Although good progress has been made, research on both human–graph and agent–graph interaction remains at an early stage, leaving many promising avenues for future work. The final part of this tutorial outlines a non–exhaustive set of open challenges.

HGI for complex data models, algorithms, and facets. To date, HGI research primarily addresses traditional graph representations (*e.g.,* property graph), creating a evident void regarding higher-order structures (*e.g.,* hypergraphs) and deep learning paradigms (*e.g.,* Graph Neural Networks). Interaction with these complex models and algorithms remain insufficiently investigated. Moreover, as remarked earlier, human interaction with graph summarization techniques remains largely uncharted.

AGI for graph data management. AGI research has paid little attention to graph data management [54, 72, 73]. At the same time, LLMs have begun reshaping data management research more broadly [38, 39]. As users increasingly delegate various tasks to agents, new frameworks are needed to ensure accurate, reliable retrieval and clear presentation of results.

Closing the loop: Human-agent interaction for graphs. We need to better integrate human and agent roles. While HGI assumes humans directly interact with graphs and AGI focuses on agents executing tasks over graph abstractions, there is limited understanding of how users should engage with agents and interpret their outputs. Bridging this gap will require leveraging HGI insights to make agent results more meaningful and usable.

Effective abstractions for interacting with large graphs. Finally, existing HGI and AGI techniques largely target small to medium-sized graphs. Scalability poses a major challenge. Future HGI work should incorporate insights from cognitive psychology and information visualization to develop improved abstractions for large schemas, queries, and datasets. Likewise, as agent tasks grow more complex, increasingly large and intricate graph abstractions will challenge both accuracy and scalability in AGI systems.

3 BIOGRAPHIES

Biographies of Sourav S Bhowmick, Arijit Khan, and Byron Choi are available at https://personal.ntu.edu.sg/assourav/about_me/biography.html, https://www.cs.bgsu.edu/arijitk/, and https://www.comp.hkbu.edu.hk/~bchoi/, respectively.

4 ACKNOWLEDGMENTS

Sourav S Bhowmick is partially supported by Ministry of Education (MOE) AcRF Tier 1 Grant No. RG19/24.

REFERENCES

[1] R. Angles, et al. PG-Schema: Schemas for Property Graphs. *Proc. ACM Manag. Data*, 1(2): 198:1-198:25, 2023.

[2] P. Anokhin, N. Semenov, A. Sorokin, D. Evseev, M. Burtsev, E. Burnaev. Arigraph: Learning Knowledge Graph World Models with Episodic Memory for LLM Agents. *In IJCAI*, 2025.

[3] D. Archambault, T. Munzner, D. Auber. GrouseFlocks: Steerable Exploration of Graph Hierarchy Space. *IEEE Trans. Vis. Comput. Graph.*, 14(4): 900-913, 2008.

[4] Y. Bei, W. Zhang, S. Wang, W. Chen, S. Zhou, H. Chen, Y. Li, J. Bu, S. Pan, Y. Yu, I. King, F. Karray, P.S. Yu. Graphs Meet AI Agents: Taxonomy, Progress, and Future Opportunities. *In arXiv:2506.18019*, 2025.

[5] M. Besta, et al. Graph of Thoughts: Solving Elaborate Problems with Large Language Models. *In AAAI*, 2024

[6] S. S. Bhowmick, B. Choi, C. Li. Graph Querying Meets HCI: State of the Art and Future Directions. In *SIGMOD*, 2017.

[7] S. S. Bhowmick, B. Choi, C. Li. Human Interaction with Graphs: A Visual Querying Perspective. *Synthesis Lectures on Data Management*, Morgan & Claypool Publishers, 2018.

[8] S. S. Bhowmick, Byron Choi. Plug-and-Play Visual Subgraph Query Interfaces. *Synthesis Lectures on Data Management*, Springer, Mar 2023.

[9] S. S. Bhowmick, Byron Choi: Data-driven Visual Query Interfaces for Graphs: Past, Present, and (Near) Future. *In SIGMOD*, 2022.

[10] S. S. Bhowmick, S. H. Annabel Chen, D. Srivastava. Cognitive Psychology Meets Data Management: State of the Art and Future Directions. *In SIGMOD*, 2024.

[11] F. Cafaro. Using Embodied Allegories to Design Gesture Suites for Human-data Interaction. *In ACM UbiComp*, 2012.

[12] X. Chang, P. Ren, P. Xu, Z. Li, X. Chen, A. Hauptmann. A Comprehensive Survey of Scene Graphs: Generation and Application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1), 2021.

[13] K. Chanthatrojwong, S. S. Bhowmick, B. Choi. LICS: Towards Theory-Informed Effective Visual Abstraction of Property Graph Schemas. *In ACM SIGMOD*, 2025.

[14] K. Chanthatrojwong, S. S. Bhowmick, B. Choi. PASCAL: A Theory-Informed Visual Interface for Property Graph Schema Visualization. *In ACM SIGMOD*, 2025.

[15] W. Chen, T. Bai, J. Su, J. Luan, W. Liu, C. Shi. KG-Retriever: Efficient Knowledge Indexing for Retrieval-Augmented Large Language Models. *In arXiv:2412.05547*, 2024.

[16] B. Chen, Z. Guo, Z. Yang, Y. Chen, J. Chen, Z. Liu, C. Shi, C. Yang. PathRAG: Pruning Graph-based Retrieval Augmented Generation with Relational Paths. *In arXiv:2502.14902*, 2025.

[17] J. M. Clark, A. Paivio. Dual coding theory and education. *Educational Psychology Review*, 3(3), 149–210, 1991.

[18] Y. Dong, X. Zhu, Z. Pan, L. Zhu, Y. Yang. Villageragent: A Graph-based Multi-agent Framework for Coordinating Complex Task Dependencies in Minecraft. *In ACL*, 2024.

[19] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, D. Metropolitansky, R. O. Ness, J. Larson. From Local to Global: A Graph RAG Approach to Query-focused Summarization. *In arXiv:2404.16130*, 2024.

[20] N. Elmqvist. Embodied Human-Data Interaction. *In ACM CHI Work. on Embodied Interact. Theory Pract.*, 2011.

[21] Z. Gao, Y. Mu, J. Qu, M. Hu, L. Guo, P. Luo, Y. Lu. DAG-Plan: Generating Directed Acyclic Dependency Graphs for Dual-arm Cooperative Planning. *In arXiv:2406.09953*, 2024.

[22] M. Gallici, M. Martin, I. Masmitja. Transfqmix: Transformers for Leveraging the Graph Structure of Multi-agent Reinforcement Learning Problems. *In AAMAS*, 2023.

[23] B. J. Gutierrez, Y. Shu, Y. Gu, M. Yasunaga, Y. Su. HippoRAG: Neurobiologically Inspired Long-term Memory for Large Language Models. *In NeuRIPS*, 2024.

[24] J. A. Guerra Gómez, A. Wilson, J. Liu, D. Davies, P. Jarvis, E. Bier. Network Explorer: Design, Implementation, and Real World Deployment of a Large Network Visualization Tool. *In AVI*, 2016.

[25] Z. Guo, L. Xia, Y. Yu, T. Ao, C. Huang. LightRAG: Simple and Fast Retrieval-augmented Generation. *In arXiv:2410.05779*, 2024.

[26] S. Han, Q. Zhang, Y. Yao, W. Jin, Z. Xu, C. He. LLM Multi-agent Systems: Challenges and Open Problems. *In arXiv:2402.03578*, 2024.

[27] X. He, Y. Tian, Y. Sun, N. Chawla, T. Laurent, Y. LeCun, X. Bresson, B. Hooi. G-retriever: Retrieval-Augmented Generation for Textual Graph Understanding and Question Answering. *In NeuRIPS*, 2024.

[28] H. Hornung, R. Pereira, M. C. C. Baranauskas, K. Liu. Challenges for Human-Data Interaction – A Semiotic Perspective, *HCI*, 1, 2015.

[29] Y. Hu, Z. Lei, Z. Zhang, B. Pan, C. Ling, L. Zhao. Grag: Graph Retrieval-augmented Generation. *In NAACL*, 2025.

[30] X. Huang, W. Liu, X. Chen, X. Wang, H. Wang, D. Lian, Y. Wang, R. Tang, E. Chen. Understanding the Planning of LLM Agents: A Survey. *In arXiv:2402.02716*, 2024.

[31] Z. Jia, J. Li, X. Qu, J. Wang, Enhancing Multi-agent Systems via Reinforcement Learning with LLM-based Planner and Graph-based Policy. *In ICRA*, 2025.

[32] J. Jiang, K. Zhou, W. X. Zhao, Y. Song, C. Zhu, H. Zhu, J.-R. Wen. KG-agent: An Efficient Autonomous Agent Framework for Complex Reasoning over Knowledge Graph. *In ACL*, 2025.

[33] A. Khan. Knowledge Graphs Querying. *SIGMOD Rec.* 52(2): 18-29, 2023.

[34] A. Khan, Y. Luo, W. Zhang, M. Zhou, X. Zhou. Retrieval-augmented Generation (RAG): What is There for Data Management Researchers? *In SIGMOD Record*, 2025.

[35] A. Khan, L. Sun, and X. Huang. LLMs+Graphs: Toward Graph-Native, Synergistic AI Systems *In PAKDD Tutorial*, 2026.

[36] E. Leurent, O.-A. Maillard, Monte-carlo Graph Search: The Value of Merging Similar States. *In Asian Conference on Machine Learning*, 2020.

[37] F. Li, P. Fang, Z. Shi, A. Khan, F. Wang, D. Feng, W. Wang, X. Zhang, Y. Cui. CoT-RAG: Integrating Chain of Thought and Retrieval-Augmented Generation to Enhance Reasoning in Large Language Models. *In EMNLP*, 2025.

[38] G. Li, J. Wang, C. Zhang, J. Wang. Data+AI: LLM4Data and Data4LLM, *In ACM SIGMOD*, 2025.

[39] C. Ma, Y. Chen, T. Wu, A. Khan, H. Wang. Unifying Large Language Models and Knowledge Graphs for Question Answering: Recent Advances and Opportunities. *In EDBT*, 2025.

[40] M. Li, S. Miao, P. Li. Simple is Effective: The Roles of Graphs and Large Language Models in Knowledge-graph-based Retrieval-augmented Generation. *In ICLR*, 2025.

[41] Y. Liu, T. Safavi, A. Dighe, D. Koutra. Graph Summarization Methods and Applications: A Survey. *ACM Comput. Surv.*, 51(3): 62:1-62:34, 2018.

[42] X. Liu, Z. Peng, X. Yi, X. Xie, L. Xiang, Y. Liu, D. Xu. Toolnet: Connecting Large Language Models with Massive Tools via Tool Graph. *In arXiv:2403.00839*, 2024.

[43] Z. Liu, Z. Lai, et al. ControlLLM: Augment Language Models with Tools by Searching on Graphs. *In ECCV*, 2024.

[44] E. Lumer, P. H. Basavaraju, M. Mason, J. A. Burke, V. K. Subbiah. Graph RAG-tool Fusion. *In arXiv:2502.07223*, 2025.

[45] L. Luo, Y.-F. Li, G. Haffari, S. Pan. Reasoning on Graphs: Faithful and Interpretable Large Language Model Reasoning. *In ICLR*, 2024.

[46] L. Luo, Z. Zhao, G. Haffari, D. Phung, C. Gong, S. Pan. GFM-RAG: Graph Foundation Model for Retrieval Augmented Generation. *In arXiv:2502.01113*, 2025.

[47] J. Ma, S. S. Bhowmick, B. Choi, L. Tay. Theories and Principles Matter: Towards Visually Appealing and Effective Abstraction of Property Graph Queries. *In SIGMOD*, 2023.

[48] J. Ma, S. S. Bhowmick, L. Tay, B. Choi. SIERRA: A Counterfactual Thinking-based Visual Interface for Property Graph Query Construction. *In SIGMOD*, 2024.

[49] T. Marzi, A. S. Khehra, A. Cini, C. Alippi. Feudal Graph Reinforcement Learning. *In Transactions on Machine Learning Research*, 2024.

[50] A. Malysheva, T. T. Sung, C.-B. Sohn, D. Kudenko, A. Shpilman. Deep Multi-agent Reinforcement Learning with Relevance Graphs. *In arXiv:1811.12557*, 2018.

[51] C. Nobre, M. D. Meyer, M. Streit, A. Lex. The State of the Art in Visualizing Multivariate Networks. *Comput. Graph. Forum*, 38(3): 807-832, 2019.

[52] S. Sahu, A. Mhedhbi, S. Salihoglu, J. Lin, M. T. Özsu. The Ubiquity of Large Graphs and Surprising Challenges of Graph Processing. *PVLDB*, 11(4), 420-431, 2017.

[53] S. Sakr, et al. The Future is Big Graphs: A Community View on Graph Processing Systems. *Commun. ACM*, 64(9): 62-71, 2021.

[54] B. Shi, I. Panagiotas. GDS Agent: A Graph Algorithmic Reasoning Agent. *In arXiv:2508.20637*, 2025.

[55] J. Sun, C. Xu, L. Tang, S. Wang, C. Lin, Y. Gong, L. Ni, H.-Y. Shum, J. Guo. Think-on-graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph. *In ICLR*, 2024.

[56] X. Tan, X. Wang, Q. Liu, X. Xu, X. Yuan, W. Zhang. Paths-Overgraph: Knowledge Graph Empowered Large Language Model Reasoning. *In ACM Web Conference*, 2025.

[57] E. Z. Victorelli, J. C. dos Reis, H. H. Hornung, A. B. Prado. Understanding Human-data Interaction: Literature Review and Recommendations for Design. *Int. J. Hum. Comput. Stud.*, 134: 13-32, 2020.

[58] C. Wang, M. Xie, S. S. Bhowmick, B. Choi, X. Xiao, S. Zhou. FERRARI: An Efficient Framework for Visual Exploratory Subgraph Search in Graph Databases. *VLDB J.*, 29(5): 973-998, 2020.

[59] L. Wang, C. Ma, et al. A Survey on Large Language Model based Autonomous Agents. *In Frontiers of Computer Science*, 18(6), 2024.

[60] Z. Wang, S. X. Teo, J. J. Chew, W. Shi. InstructRAG: Leveraging Retrieval-augmented Generation on Instruction Graphs for LLM-based Task Planning. *In SIGIR*, 2025.

[61] Y. Wen, Z. Wang, J. Sun. Mindmap: Knowledge Graph Prompting Sparks Graph of Thoughts in Large Language Models. *In ACL*, 2024.

[62] X. Wen, Q. Fu, S. Han, Y. Guo, J. K. Liu, Y. Wang. Envisage: Towards Expressive Visual Graph Querying. *In IEEE VIS*, 2025.

[63] J. M. Wolfe, I. S. Utochkin. What is a preattentive feature? *Current Opinion in Psychology*, 29, 19–26, 2019.

[64] M. Wooldridge, N. R. Jennings. Intelligent agents: Theory and practice. *The knowledge engineering review*, 10(2), 1995.

[65] Y. Wu, Y. Fan, S. Y. Min, S. Prabhumoye, S. M. McAleer, R. Salakhutdinov, Y. Bisk, Y. Li, T. Mitchell. Agentkit: Structured LLM Reasoning with Dynamic Graphs. *In the First Conference on Language Modeling*, 2024.

[66] X. Wu, Y. Shen, C. Shan, K. Song, S. Wang, B. Zhang, J. Feng, H. Cheng, W. Chen, Y. Xiong. Can Graph Learning Improve Planning in LLM-based Agents? *In NeurIPS*, 2024.

[67] W. Xu, K. Mei, H. Gao, J. Tan, Z. Liang, Y. Zhang. A-MEM: Agentic Memory for LLM Agents. *In arXiv:2502.12110*, 2025.

[68] H. Yang, J. Chen, M. Siew, T. Lorido-Botran, C. Joe-Wong. LLM-powered Decentralized Generative Agents with Adaptive Hierarchical Knowledge Graph for Cooperative Planning. *In arXiv:2502.05453*, 2025.

[69] Y. Yao, Z. Li, H. Zhao. Got: Effective Graph-of-thought Reasoning in Language Models. *In NAACL*, 2024.

[70] V. Yoghourdjian, D. Archambault, S. Diehl, T. Dwyer, K. Klein, H. C. Purchase, H.-Y. Wu. Exploring the Limits of Complexity: A Survey of Empirical Studies on Graph Visualization. *Visual Informatics* 2(4), 2018.

[71] V. Yoghourdjian, T. Dwyer, K. Klein, K. Marriott, M. Wybrow. Graph Thumbnails: Identifying and Comparing Multiple Graphs at a Glance. *IEEE Trans. Vis. Comput. Graph.*, 24(12): 3081-3095, 2018.

[72] Z. Yuan, M. Liu, H. Wang, B. Qin. MA-GTS: A Multi-agent Framework for Solving Complex Graph Problems in Real-world Applications. *In CoRR*, abs/2502.18540, 2025.

[73] X. Zhao, M. Blum, F. Gao, Y. Chen, B. Yang, L. Marquez-Carpintero, M. Pina-Navarro, Y. Fu, S. Morikawa, Y. Iwasawa, Y. Matsuo, C. Park, I. Li. Agentigraph: A Multi-agent Knowledge Graph Framework for Interactive, Domain-specific LLM Chatbots. *In CoRR,* abs/2508.02999, 2025.

[74] R. Zhao, F. Zhao, L. Wang, X. Wang, G. Xu. KG-COT: Chain-of-thought Prompting of Large Language Models over Knowledge Graphs for Knowledge-aware Question Answering. *In IJCAI*, 2024.

[75] M. Zhuge, W. Wang, L. Kirsch, F. Faccio, D. Khizbullin, J. Schmidhuber. GPTSwarm: Language Agents as Optimizable Graphs. *In ICML*, 2024.