

# StructuGraphRAG: Structured Document-Informed Knowledge Graphs for Retrieval-Augmented Generation

Xishi Zhu<sup>1\*</sup>, Xiaoming Guo<sup>1\*</sup>, Shengting Cao<sup>1</sup>, Shenglin Li<sup>1</sup>, Jiaqi Gong<sup>1†</sup>

<sup>1</sup>Department of Computer Science, University of Alabama, Tuscaloosa, Alabama, USA  
xzhu39@crimson.ua.edu, xguo29@crimson.ua.edu, scao7@crimson.ua.edu, sli90@crimson.ua.edu, jiaqi.gong@ua.edu

## Abstract

Retrieval-augmented generation (RAG) enhances large language models (LLMs) by incorporating external data sources beyond their training sets and querying predefined knowledge bases to generate accurate, context-rich responses. Most RAG implementations use vector similarity searches, but the effectiveness of this approach and the representation of knowledge bases remain underexplored. Emerging research suggests knowledge graphs as a promising solution. Therefore, this paper presents StructuGraphRAG, which leverages document structures to inform the extraction process and constructs knowledge graphs to enhance RAG for social science research, specifically using NSDUH datasets. Our method parses document structures to extract entities and relationships, constructing comprehensive and relevant knowledge graphs. Experimental results show that StructuGraphRAG outperforms traditional RAG methods in accuracy, comprehensiveness, and contextual relevance. This approach provides a robust tool for social science researchers, facilitating precise analysis of social determinants of health and justice, and underscores the potential of structured document-informed knowledge graph construction in AI and social science research.

## Introduction

Retrieval-augmented generation (RAG) (Lewis et al. 2020) is a promising technique for enhancing the capabilities of large language models (LLMs) (Wu et al. 2024), particularly when addressing queries that require precise and accountable information beyond what can be provided by probability predictions, such as the general next-token prediction paradigm (Gloeckle et al. 2024). However, a significant challenge in implementing RAG for LLMs lies in effectively representing knowledge and retrieving information (Ding et al. 2024; Wiratunga et al. 2024). The representation aspect involves structuring knowledge to accurately capture entities and their interrelationships, which is crucial for retrieving precise and contextually relevant information. On the other hand, the retrieval process entails dynamically querying this structured knowledge base to supply LLMs with accurate

and up-to-date information. By focusing on both representation and retrieval, RAG can significantly improve the quality and relevance of generated responses, thereby enhancing the overall performance of LLMs in delivering accountable and context-rich answers.

Recent advancements in integrating knowledge graphs (KGs) with RAG (Sanmartin 2024) have shown significant improvements in information retrieval and response generation for LLMs. In the domain of customer service question answering, a novel method constructs KGs from historical issues (Vekariya and Limbasiya 2020), preserving both intra-issue and inter-issue relationships. This method enhances retrieval accuracy and response quality by mitigating text segmentation challenges that are common in traditional RAG approaches. Another study (Sawarkar, Mangal, and Solanki 2024) benchmarked the effectiveness of graph-based versus vector-based search methods in financial report analysis, demonstrating that KGs provide a comprehensive view of company performance by structuring earnings call transcripts into relationships among financial metrics, products, and market conditions. Additionally, combining KGs with traditional vector-based RAG allows for more precise navigation through data layers (Hussien et al. 2024), facilitating both deep and broad information retrieval, thereby enhancing the granularity and relevance of the retrieved information. These studies highlight the potential of KGs to overcome the limitations of vector similarity searches in RAG, offering a structured framework that captures complex interrelationships and improves the accuracy and contextual relevance of LLM-generated responses across various applications.

The primary motivation for our study is to investigate the capacities of LLMs in understanding critical social science problems and societal needs, such as social determinants of health and justice, and in delivering accountable information and knowledge pertinent to the communities and regions in need. This endeavor aims to enhance public health awareness and inform policy-making processes. Social science research and data are often organized in complex, multi-dimensional formats that encompass a diverse range of variables and relationships. These datasets typically include detailed codebooks that define variables, describe their possible values, and explain their interconnections. For instance, datasets like the National Survey on Drug Use and Health

\*These authors contributed equally.

†Corresponding author.

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

(NSDUH) and the National Incident-Based Reporting System (NIBRS) contain rich metadata, covering demographic characteristics, behavioral patterns, health outcomes, and crime incidents. This inherent structure provides a comprehensive context essential for understanding the relationships between various social determinants and their impacts on health and justice outcomes.

The significance of providing accountable information through LLMs in this context is profound. Accurate and contextually relevant responses are indispensable for researchers, policymakers, and public health officials who depend on this information to make informed decisions. By leveraging the detailed structures of social science datasets, LLMs can generate responses that are not only precise but also grounded in a robust contextual framework. This ensures that the information provided is reliable, verifiable, and actionable. In areas such as public health and justice, where decisions based on inaccurate or incomplete data can have severe consequences (Hager et al. 2024), the importance of accountable information cannot be overstated. Enhancing the accuracy and relevance of information retrieval and generation through our approach supports better decision-making. Ultimately, this contributes to more effective policies and interventions that address the complex social determinants of health and justice.

Our method stands out by utilizing LLMs, derived multi-agents systems, and prompt engineering to extract KGs from meta-datasets based on previous social science studies and data repositories, including detailed codebook structures, and then implementing the extracted KGs for RAG of LLMs. These datasets contain rich contextual information about entities and their relationships, which significantly informs the KG implementation process. By leveraging this contextual richness, our approach enhances both the graph-based and vector-based search processes within RAG. This methodology enables more accurate and contextually relevant entity and relationship extraction, facilitating nuanced and comprehensive information retrieval. Integrating these enriched KGs with RAG allows LLMs to generate detailed and contextually appropriate responses, addressing the limitations of traditional vector similarity searches and supporting more informed decision-making in social science research and public health policy.

Our work makes several significant contributions to the field.

- **Structured Document-Informed Knowledge Graph Construction:** we proposed a novel approach for constructing structured document-informed knowledge graphs by leveraging the capabilities of LLMs fine-tuning. This method enhances the understanding and application of social science domain knowledge within LLMs, enabling them to process and integrate complex, multi-dimensional data effectively.
- **Innovative Model with Multi-Agents and Prompt Engineering:** We propose an innovative model that utilizes the advanced capabilities of LLMs, including the use of multi-agents and prompt engineering. This model significantly improves the effectiveness and efficiency of struc-

tured entity, and relation extraction, making it more precise and contextually relevant.

- **Synergy Between LLMs and Knowledge Graphs:** We demonstrate the powerful synergy between LLMs and knowledge graphs (KGs) in the social science domain. This synergy is particularly effective in delivering and answering critical and accountable public health and justice information, showcasing the potential of LLMs and KGs to provide reliable, actionable insights for researchers and policymakers.

## Related Work

Recent research highlights the significant advancements achieved by integrating KGs with LLMs (Ding et al. 2024), demonstrating substantial improvements in various applications such as information retrieval (Wu et al. 2023), question answering (Feng, Zhang, and Fei 2023; Pan et al. 2024), and content generation (Meyer et al. 2023). KGs provide structured, contextual knowledge that enhances the accuracy and relevance of LLM outputs, making them particularly effective in complex domains like biomedicine and finance. One notable approach is the use of "chain-of-thought prompting," (Wei et al. 2022) where LLMs utilize a chain-like graph structure to enhance their reasoning capabilities. This method has been shown to significantly improve performance in tasks like math word problems and logical reasoning by providing a structured sequence of steps that LLMs can follow to arrive at a solution. More advanced methods, such as "Tree of Thoughts" (ToT) (Yao et al. 2024) and "Graph of Thoughts" (GoT) (Besta et al. 2024), utilize tree and graph structures to facilitate more complex reasoning processes, demonstrating substantial improvements over simpler models. ToT employs tree structures to represent different possible reasoning paths, allowing LLMs to explore multiple avenues before converging on the most accurate solution. GoT extends this further by using heterogeneous graph structures that integrate diverse nodes and complex interactions, enhancing the model's ability to handle multifaceted tasks. These advancements highlight the transformative potential of combining KGs with LLMs, enabling more sophisticated and reliable AI applications.

LLMs are being utilized to automate the construction of KGs from unstructured data by extracting entities and relationships and structuring them into triples (Safavi and Koutra 2021). Techniques, such as few-shot learning (Patil and Gudivada 2024; Song et al. 2023), help train LLMs to create detailed and accurate KGs, capturing complex data relationships. Recent advancements focus on enhancing quality and scalability, optimizing token usage, and ensuring the relevance of extracted information, thereby improving applications like fact-checking and domain-specific question answering (Zhang et al. 2024). One notable implementation is OntoGPT (Caufield et al. 2024; Matentzoglou et al. 2023), which leverages GPT-3 to efficiently extract domain-specific concepts and relationships from unstructured text. OntoGPT uses a two-stage pipeline: a fine-tuned GPT-3 model identifies potential ontological elements, and then a rule-based system refines these elements into a formal ontology. This

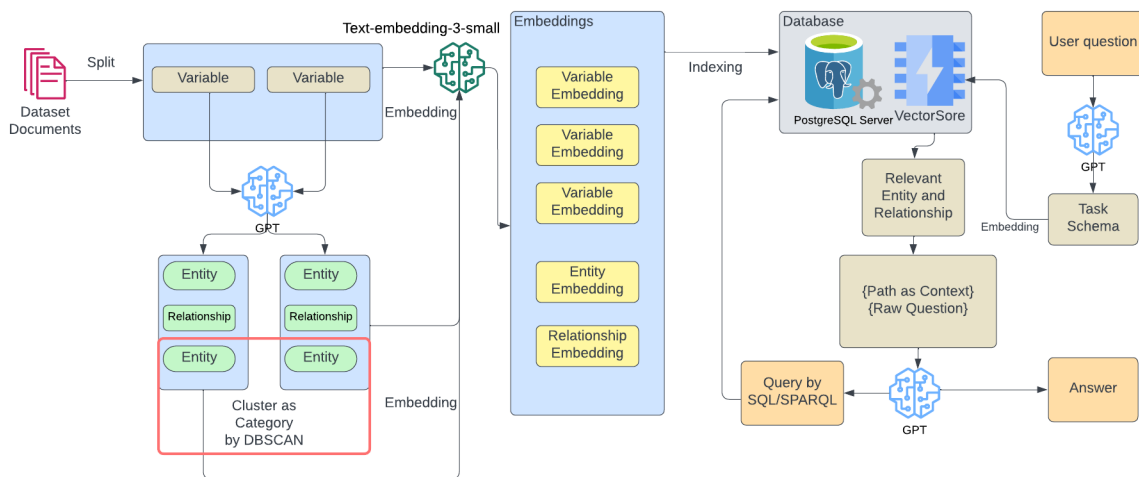


Figure 1: Overview of the StructuGraphRAG workflow. The process begins with parsing the data codebook into structured chunks, followed by fine-tuning GPT for entity and relation extraction. Extracted elements are converted to embeddings and clusters. Entities are then mapped to codebook classes, and a knowledge graph is constructed incorporating survey responses. When a user query is received, it’s embedded and used to retrieve relevant information from the knowledge graph. Finally, GPT generates a response based on this retrieved-context, providing accurate and contextually appropriate answers to user queries about substance use and mental health data.

approach reduces the time required for ontology construction by 40% compared to manual methods while maintaining high quality. However, OntoGPT faces challenges such as biases from GPT-3’s training data, difficulty with specialized concepts, and dependency on the quality of the seed ontology. These limitations highlight the need for minimal domain expert involvement in automated ontology construction, focusing experts primarily on evaluation.

Integrating KGs into RAG systems for LLMs significantly enhances information retrieval and response generation (Procko 2024). KGs offer a structured framework that guides the retrieval process, improving precision and recall compared to traditional vector-based methods. This structured approach is particularly effective in tasks such as question answering and information summarization, where the contextual relevance of retrieved information is crucial. Additionally, KGs enrich the generation phase by providing additional context and knowledge (Chen et al. 2024), resulting in more informative and coherent outputs. An exemplary implementation of this approach is Microsoft’s GraphRAG (Edge et al. 2024). GraphRAG leverages KGs to structure and organize knowledge, enhancing LLMs’ retrieval and generation capabilities. It constructs a knowledge graph from input text, capturing entities and their relationships, which is then used to improve query responses. This method addresses the limitations of traditional RAG approaches (Cuconasu et al. 2024; Hu and Lu 2024) by utilizing the structured nature of KGs to provide contextually rich and comprehensive answers. The effectiveness

of GraphRAG, like other KG-based systems, hinges on the completeness and accuracy of the KGs, underscoring the importance of robust KG construction and maintenance, which motivated our work in deep investigating the structured documents in some specific domains, such as social science knowledge bases and data repositories.

## Approach

In this section, we first introduce the dataset adopted for this study. We then present an overview of our proposed StructuGraphRAG workflow and compare it to the traditional GraphRAG framework design.

### Dataset

For this study, we selected the 2022 National Survey on Drug Use and Health (NSDUH) from the Substance Abuse and Mental Health Services Administration (SAMHSA). The NSDUH is a widely recognized dataset in substance use disorders (SUDs) research, encompassing a broad spectrum of factors associated with substance use, including substance use patterns, treatment, mental health issues, and demographics. We obtained two types of public use data from the provider:

- **Data Codebook:** a guide to the NSDUH public use data file. It provides the variable name, a description of the variable, an answer value code, and an answer value description. Notably, the codebook provides a domain-specific class hierarchy that categorizes variables within

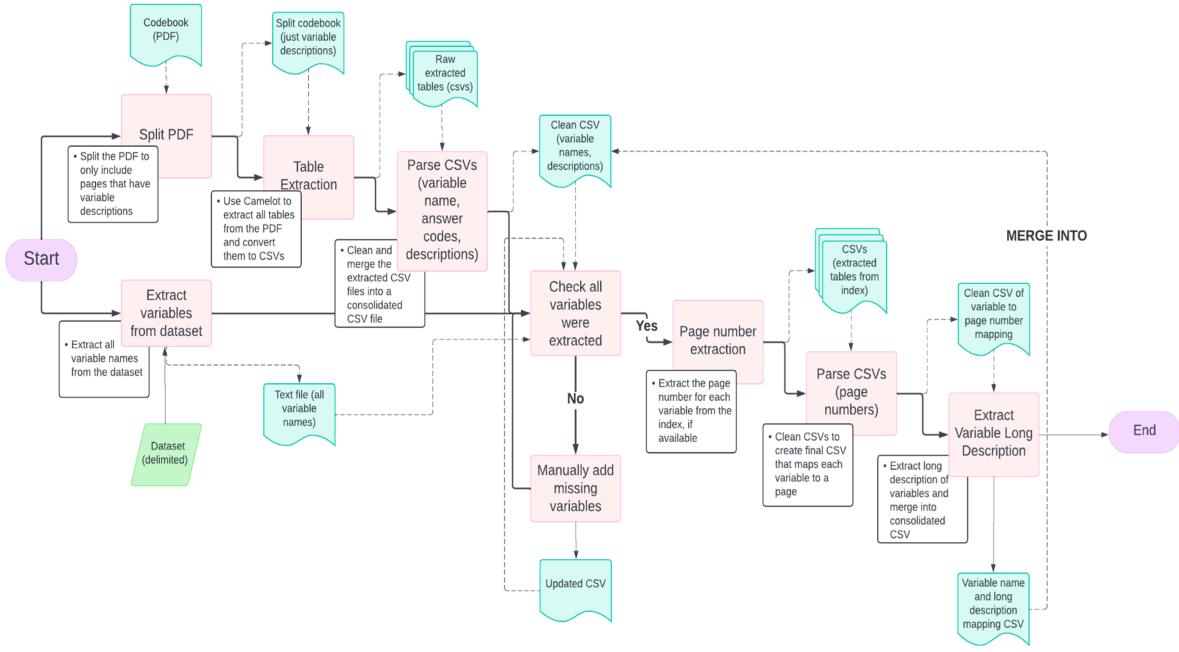


Figure 2: Workflow overview of codebook PDF parsing.

the survey, offering valuable domain context for data interpretation.

- **Raw Survey Response:** raw tabular data collected through NSDUH survey. Each column in the table represents a variable code, corresponding to a specific survey question detailed in the codebook. Rows represent individual participant responses, with integer values that map to answer meanings as defined in the codebook.

## GraphRAG vs StructuGraphRAG

In this section, we first provide a brief overview of the traditional GraphRAG workflow. We then present our proposed StructuGraphRAG approach, detailing its workflow and highlighting how it improves upon traditional GraphRAG methods in knowledge graph construction. We explain the key innovations of our approach, particularly in leveraging structured documents to enhance the accuracy and relevance of the resulting knowledge graphs for social science research.

**GraphRAG** GraphRAG enhances traditional RAG by incorporating knowledge graph structures into the retrieval process. The workflow typically begins with document pre-processing, where texts are segmented into chunks using methods such as fixed-size windowing or sentence-based splitting. Entity recognition and relation extraction algorithms, often leveraging pre-trained language models or rule-based systems, are then applied to these chunks to identify key concepts and their relationships. The results of this process are used to construct a knowledge graph, where the extracted entities serve as nodes and the identified relationships form the edges. During query processing, GraphRAG

employs a dual-retrieval approach: it performs both textual similarity search on the original chunks and graph traversal operations on the constructed knowledge graph, often using techniques like graph neural networks or random walks to capture structural information. This approach addresses the limitations of text-only RAG by capturing semantic relationships more effectively, enabling multi-hop reasoning through graph structures, and providing more coherent and contextually relevant information for complex queries that require an understanding of interconnected concepts.

However, GraphRAG faces several limitations, particularly in processing structured documents common in specialized domains like social science research. The arbitrary chunking of documents can lead to fragmented or decontextualized information in the resulting knowledge graph. The graph construction process often lacks domain-specific structural guidance, relying primarily on general-purpose NLP techniques that may not capture accurate hierarchies and relationships present in specialized documents. This can result in missing crucial hierarchical relationships that are inherent in structured documents like codebooks or survey instruments. Additionally, the quality of responses heavily depends on the accuracy and completeness of the initial knowledge graph construction, which can be error-prone and computationally expensive. Balancing the influence of graph-based and text-based retrieval in the final context selection can also be complex, potentially leading to suboptimal performance if not carefully tuned.

We implement GraphRAG in our study as a baseline for performance comparison with StructuGraphRAG. We leverage the GeneralAgent(Li 2024) framework alongside GraphRAG to autonomously execute complex tasks. As

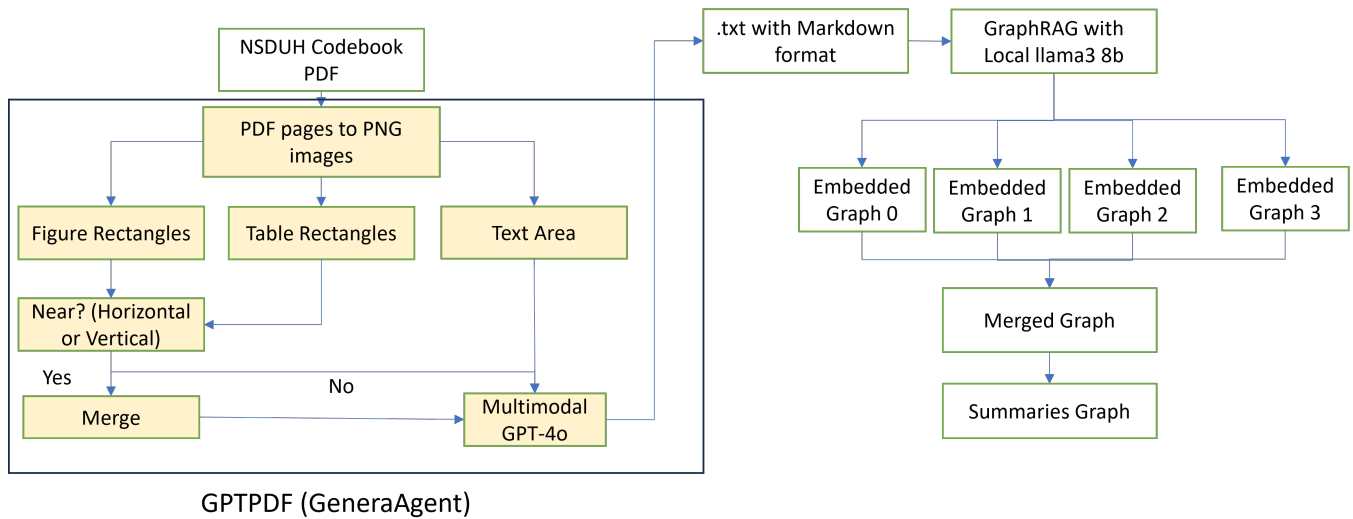


Figure 3: Flowchart for our GraphRAG implementation with GeneralAgent tool

shown in Fig 3, the process begins with the NSDUH Codebook PDF, which is converted into a series of PNG images, one for each page. These images are analyzed to identify and extract figure and table rectangles as well as text areas. The figure rectangles are further examined to determine their proximity (whether they are horizontally or vertically near) and then merged accordingly. Similarly, table rectangles are processed to form complete tables. The text areas, along with the merged figures and tables, are parsed using a multimodal GPT-4 model to convert the content into a structured Markdown file.

The next stage involves feeding the Markdown text into the GraphRAG framework, where it interacts with a local instance of the GPT 4o model. This framework generates four separate embedding graphs from the input text. These graphs are then merged to form a single comprehensive graph, which is further distilled into a summary graph, capturing the essential information. This entire workflow, illustrated in the provided figure, ensures that complex PDF documents are effectively parsed, analyzed, and summarized using advanced multimodal and embedding techniques.

**StructuGraphRAG** Our proposed StructuGraphRAG aims to overcome inherent limitations from GraphRAG such as arbitrary text chunk segmentation and lack of domain-specific structural guidance in graph construction. The workflow overview of our proposed StructuGraphRAG is shown in Fig 1.

**Codebook Parsing** The initial step in our StructuGraphRAG approach involves parsing the codebook, a critical process that forms the foundation for our knowledge graph construction and overcomes the decontextualized text chunk issue in GraphRAG. We designed the parsing process to extract and preserve the inherent structure of the codebook, ensuring that each piece of chunk retains only one variable and their related descriptions, as illustrated in Fig 2.

We begin by splitting the PDF codebook into individual pages to facilitate granular processing. Using the Camelot library, a powerful tool for PDF table extraction, we systematically process each page to extract tables containing variable descriptions. The extracted tables are then converted into CSV (Comma-Separated Values) format, which allows for easier manipulation and analysis of the data. This conversion step is essential as it transforms the semi-structured PDF content into a fully structured format that can be programmatically processed.

Following the table extraction, we perform a series of data-slicing and validation steps:

- We parse the table to extract variable page numbers in the codebook and obtain the variable descriptions.
- A comprehensive check is conducted to ensure all variables from the dataset are successfully extracted from the codebook.
- In cases where variables are missing, we implement a manual addition process to ensure completeness.
- For variables that are successfully extracted, we further process them to include page number information, which aids in maintaining traceability back to the original document.

The final step in our codebook parsing workflow involves merging all the processed information. We combine the cleaned CSV tables, the page number mappings, and the extracted long descriptions for each variable. This merged dataset represents a comprehensive, structured representation of the codebook, capturing not just the variable definitions but also their context within the larger survey structure.

**Entity and Relation Extraction using Multi-agent LLMs Fine-tuning** After getting the chunk slices from the codebook, we employed a multi-stage process that leverages the contextual chunk slices and fine-tunes LLMs to generate a comprehensive list of entity and relation pairs to form an

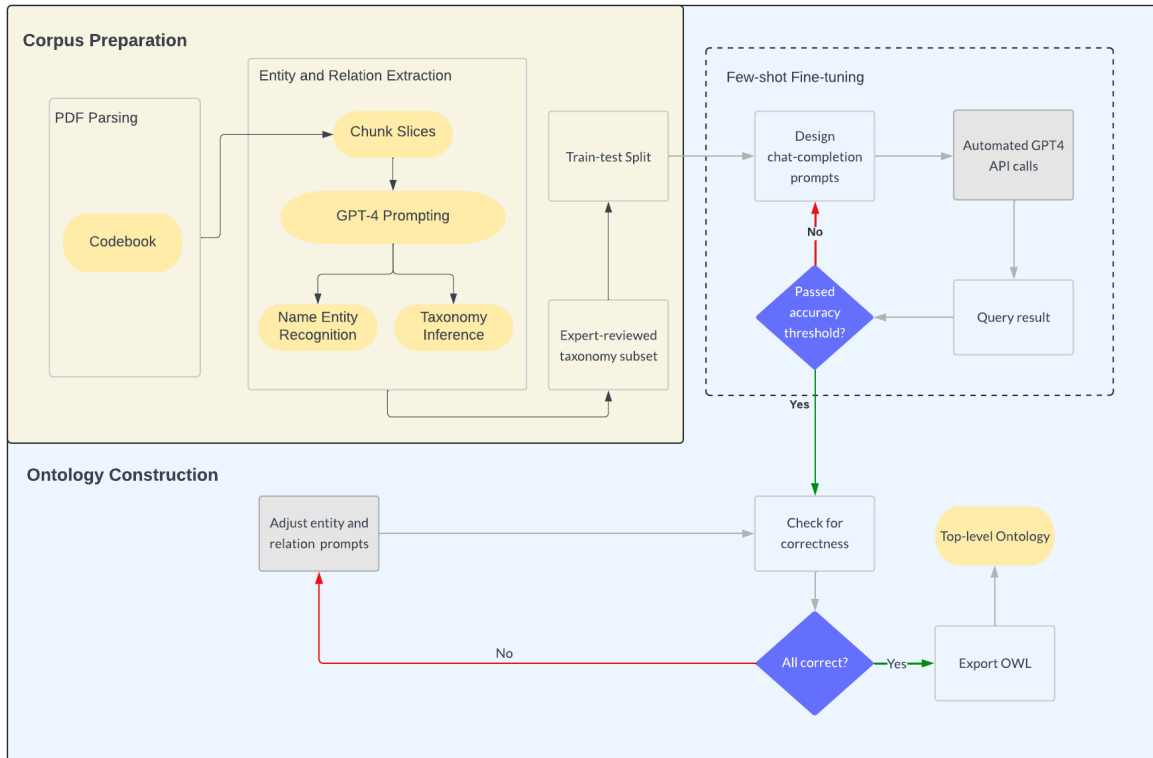


Figure 4: Workflow overview of entity and relation extraction using multi-agent LLMs Fine-tuning.

ontology(Fig 4). This approach is designed to overcome the limitations of traditional ontology construction methods by utilizing the LLMs as agents to extract entities and relations from domain-specific text directly. The two-stage process is:

**Corpus preparation:** at this stage we aim to gather contextual information from the data source provider and transform it into GPT-usable training data, and the process is described below.

- **PDF Parsing:** We process the data codebook, transforming it from PDF format into structured text chunks and align with variables.
- **GPT-4 Prompt Engineering:** find the optimal format to feed GPT with entity and relation extraction example from the text chunk. (During the past 12 months, have you driven a vehicle while you were under the influence of alcohol? – Person + drive under influence + alcohol).
- **Training Samples:** prepare entity and relation extraction examples for fine-tuning.

**Ontology Construction:** this stage we will leverage the prepared example to fine-tune the GPT model and populate to the whole dataset and generate the ontology.

- **Fine-tuning:** prepare a GPT model for fine-tuning and submit fine-tuning job with prepared training samples.
- **Evaluation:** evaluate the performance of a fine-tuned model and check for correctness rate from generated entities and relations.

- **Ontology Generation:** map the identified entity to the class defined by the domain expert in the codebook, and generate the ontology for the whole dataset.

**Knowledge Graph Construction** Building upon the robust entity and relation extraction process, we employ PostgreSQL, a powerful relational database management system, to design and implement an Entity-Relationship Diagram (ERD) that captures the complex structure of our extracted entities and their relationships. Each entity type is represented as a table, with attributes corresponding to its characteristics as identified in the codebook. Relationships between entities are implemented through foreign key constraints, ensuring data integrity and enabling efficient querying.

Once the database schema is established, we populate it with two types of data:

- **Extracted Entities and Relationships:** The entities and relationships identified through our GPT fine-tuning process are inserted into their respective tables. This forms the structural backbone of our knowledge graph.
- **Survey Response Data:** We then fetch the actual survey response data and associate it with the corresponding entities in the database. This step is crucial as it transforms our conceptual graph into a data-rich knowledge representation.

	Number of Classes	Number of Properties	Class Richness	Property Richness	Inheritance Richness
GraphRAG	1	10	0	0	0
StructuGraphRAG	1256	98	1	0.06	1.48
Rural KG(manual)	9	44	1	4.56	1.80

Table 1: Intrinsic evaluation results using three different methods

**LLM-based Knowledge Retrieval** The LLM-based knowledge retrieval process consists of three phases: locating the entities in the vector index, retrieving the original text for prompt engineering, and feeding the retrieved data, along with the raw question, into the LLM model. During the entity location task, the retrieved entities may vary—they can follow a long path, reside within the same cluster or community, or represent just a single entity. This variation depends on the balance between performance and cost.

Embedding is the most commonly used method for entity location. In our scheme, after categorizing the client request into a predefined domain, the entity location task is performed on a category of entities, such as substances or incidents. The client’s query is converted into an embedding vector, and then similarity or distance calculations are performed with the entity embeddings in the search scope, ensuring the most relevant entities are selected. Reference identifiers for the entities are then returned for the retrieval of literal descriptions.

Original literal descriptions serve as the context for the prompts. The similarity of the embeddings ensures that they are relevant to the objective underlying the client’s question, effectively bridging the question to the local corpus. Consequently, the potential knowledge unveiled through this method is more aligned with the user’s needs and the specificities of the query domain. This process not only enhances the precision of the retrieved information but also helps uncover deeper insights that might not be immediately apparent from the question alone, making the LLM-based knowledge retrieval system robust and efficient in handling a wide range of queries with varying degrees of complexity.

For numeric queries, group search can be predefined due to our coherent and clearly defined categories. After locating the entities through embedding matching, the query statement can be switched between multiple predefined templates, ensuring that accurate numeric queries are reliably handled.

## Experiments

In this study, we conducted a comprehensive evaluation of the ontology quality produced by the structuGraphRAG method. The evaluation aims to enable us to examine both the internal consistency and logical coherence of the ontology, as well as its practical applicability and performance in real-world query-answer scenarios. Our assessment approach comprised two distinct phases: intrinsic and extrinsic

evaluation.

### Intrinsic Evaluation

The intrinsic evaluation phase focused on assessing the ontology’s internal structure and logical integrity. We utilized HermiT Reasoner (Harrell 1946) to conduct this evaluation, and calculated several evaluation metrics to quantify the ontology’s quality:

- Number of classes: a straightforward count of all classes in the ontology.
- Number of properties: a count of all properties in the ontology.
- Class richness: this metric is defined as the ratio of the number of classes to the total number of entities (classes and individuals). A higher class richness indicates a well-defined and detailed conceptualization of the domain, suggesting a rich set of concepts.
- Property richness: calculated as the ratio of the number of properties to the total number of entities. Higher property richness implies a diverse and comprehensive set of relationships and attributes, indicating the ontology’s ability to capture detailed information about entities and their interactions.
- Inheritance richness: this is measured as the average number of subclasses per class. Higher inheritance richness suggests a well-structured and deep class hierarchy, indicating a more detailed and hierarchically organized representation of the domain.

### Extrinsic Evaluation

The extrinsic evaluation of the StructuGraphRAG method focuses on assessing its performance in real-world query-answering scenarios. This evaluation aims to compare the effectiveness of our ontology-enhanced approach against other state-of-the-art language models, specifically GPT-4o and standard GraphRAG. The evaluation protocol involves the development of a comprehensive set of domain-specific queries, designed to test various aspects of knowledge representation and reasoning capabilities. These queries will be processed through the structuGraphRAG system, as well as through GPT-4o and standard GraphRAG for comparative analysis. The qualitative assessment criteria include response accuracy, relevance, completeness, consistency with domain knowledge, and depth of reasoning. We expect that the StructuGraphRAG method will demonstrate enhanced



GraphRAG	ChatGPT 4o	StructuGraphRAG
<p>Q: In the National Survey on Drug Use and Health (NSDUH) 2022 dataset, what is the most commonly used kind of drug in vehicle-related illegal activities?</p> <p>A: In summary, <b>alcohol</b> is the most commonly associated drug with vehicle-related illegal activities according to the 2022 NSDUH dataset. However, other substances like <b>cocaine</b>, <b>hallucinogens</b>, <b>heroin</b>, and <b>marijuana</b> also contribute to the prevalence of <b>DUI offenses</b>. The data underscores the critical need for continued efforts to address and mitigate the risks associated with impaired driving [Data: Reports (102, 71)].</p>	<p>Q: In the National Survey on Drug Use and Health (NSDUH) 2022 dataset, what is the most commonly used kind of drug in vehicle-related illegal activities?</p> <p>A: In the 2022 National Survey on Drug Use and Health (NSDUH), <b>marijuana</b> was the drug most commonly associated with <b>vehicle-related illegal activities</b>. The survey indicated that marijuana use was frequently reported among individuals involved in such activities, particularly when it came to <b>driving under the influence (DUI)</b>. This aligns with broader trends observed in substance use data, where marijuana continues to be one of the most widely used illicit drugs in the United States.</p> <p>The survey highlighted that the perception of risk related to driving after marijuana use has been relatively low among users, which may contribute to its prevalence in vehicle-related offenses. Additionally, the survey provided insights into how various demographic groups differ in their patterns of drug use and associated illegal activities. For more detailed information, you can refer to the official NSDUH report and related summaries.</p>	<p>Q: In the National Survey on Drug Use and Health (NSDUH) 2022 dataset, what is the most commonly used kind of drug in vehicle-related illegal activities?</p> <p>A: In the 2022 National Survey on Drug Use and Health (NSDUH) dataset, <b>alcohol</b> is the drug most commonly associated with vehicle-related illegal activities. The provided statistics indicate that <b>alcohol</b> has the highest count (2692) among the listed substances in relation to <b>vehicle-related illegal activities</b>. This is followed by <b>marijuana</b> with a count of 2598. Given the significant difference in counts between alcohol and other substances, it is reasonable to conclude that alcohol is the most commonly associated drug with vehicle-related illegal activities in the 2022 NSDUH dataset.</p> <p>Ontology Class:</p> <ul style="list-style-type: none"> <li>Substance</li> </ul> <p>Relationship:</p> <ul style="list-style-type: none"> <li><b>Driving Under the Influence of Substances</b></li> </ul> <p>KG entity:</p> <ul style="list-style-type: none"> <li><b>Marijuana</b></li> <li><b>Alcohol</b></li> </ul>

Figure 5: Model response comparison for query on commonly used drugs in vehicle-related illegal activities.

GraphRAG	ChatGPT 4o	StructuGraphRAG
<p>Q: In the 2022 National Survey on Drug Use and Health (NSDUH) dataset, what is the kind of drug least used in the prescription drug misuse incidents?</p> <p>A: In summary, <b>Phendimetrazine</b> and <b>Daytrana</b> are the least used drugs in prescription drug misuse incidents according to the 2022 NSDUH dataset, each with only one reported case of misuse. Other drugs such as <b>Benzphetamine</b>, <b>Dexedrine</b>, and <b>Focalin</b> also show very low misuse rates, highlighting the varied landscape of <b>prescription drug misuse</b> [Data: Reports (283, 287, 282, 281, 285, +more)].</p>	<p>Q: In the 2022 National Survey on Drug Use and Health (NSDUH) dataset, what is the kind of drug least used in the prescription drug misuse incidents?</p> <p>A: To identify the least used drug in prescription drug misuse incidents within the 2022 NSDUH dataset, you would need to analyze the relevant sections of the dataset that capture information about prescription drug misuse. Typically, this data is recorded in variables or columns that specify the type of drug misused, such as opioids, <b>benzodiazepines</b>, <b>stimulants</b>, etc.</p> <p>Here is a general approach to find the least used drug:</p> <p>Load the Dataset: Import the NSDUH dataset into a data analysis tool like Python (using pandas) or R.</p> <p>Identify Relevant Columns: Locate the columns that track the type of drugs involved in misuse incidents. These columns might be labeled with names indicating specific drug types or a general category like "Prescription Drug Misuse".</p> <p>Aggregate Data: Count the occurrences of each drug type in the prescription drug misuse incidents.</p> <p>Determine the Least Used Drug: Sort the counts to identify the drug with the lowest usage frequency.</p>	<p>Q: In the 2022 National Survey on Drug Use and Health (NSDUH) dataset, what is the kind of drug least used in the prescription drug misuse incidents?</p> <p>A: In the 2022 National Survey on Drug Use and Health (NSDUH) dataset, the least used drug in the prescription drug misuse incidents is <b>Sedative</b> with a count of 146. To determine the least used drug in <b>prescription drug misuse incidents</b>, I analyzed the provided statistics, focusing on the 'count' field for each substance. By sorting the substances based on their counts in ascending order, I identified <b>Sedative</b> as the substance with the lowest count (146), making it the least used drug in the dataset. The other substances had higher counts: <b>Pain Reliever</b> (1523), <b>Stimulant</b> (984), and <b>Tranquilizer</b> (810).</p> <p>Ontology Class:</p> <ul style="list-style-type: none"> <li>Substance</li> </ul> <p>Relationship:</p> <ul style="list-style-type: none"> <li><b>Prescription Drug Misuse</b></li> </ul> <p>KG entity:</p> <ul style="list-style-type: none"> <li><b>Sedative</b></li> <li><b>Tranquilizer</b></li> <li><b>Stimulant</b></li> <li><b>Pain Reliever</b></li> </ul>

Figure 6: Model response comparison for query on least used drugs in prescription drug misuse incidents.



performance due to its integration of domain-specific ontology and knowledge graph. Also, StructuGraphRAG enables the system to decompose input queries into constituent entities and relations, facilitating precise matching with the ontological structure and subsequent retrieval of relevant data.

## Results

The intrinsic evaluation results, summarized in Table 1, reveal substantial differences in ontological structure among the three methods examined. GraphRAG generated a very simple class structure consisting of a single class and 10 properties, resulting in null values across all richness metrics. In contrast, we observed that StructuGraphRAG produced a significantly more complex ontology, comprising 1256 classes and 98 properties. This resulted in a class richness of 1, property richness of 0.06, and inheritance richness of 1.48. The manually constructed knowledge graph (RuralKG), while more compact with 9 classes and 44 properties, achieved the highest property richness (4.56) and inheritance richness (1.80) among the three approaches.

The results demonstrated that StructuGraphRAG is able to generate a more comprehensive and well-structured ontology. We believe this can be attributed to the integration of domain knowledge structure from the codebook, and we believe that the lack of domain knowledge guidance in ontology construction process is the main reason causing the GraphRAG failed to capture the class hierarchy. While the Rural KG demonstrates higher property and inheritance richness, indicating a densely connected and deeply hierarchical structure relative to its compact size, StructuGraphRAG’s extensive class hierarchy provides a broader foundation for representing complex domains. This balance between breadth (as evidenced by the number of classes) and depth (reflected in the positive inheritance richness) positions StructuGraphRAG as an effective tool for automatically generating rich, and relatively comprehensive ontologies.

The extrinsic evaluation results showed that StructuGraphRAG outperformed others in addressing domain-specific queries. Figure 5 illustrates responses to the query “commonly used kind of drug in vehicle-related illegal activities.” While GraphRAG and GPT 4o captured the query’s essence, identifying driving under the influence (DUI) and alcohol as relevant factors, it is hard to understand how this knowledge was retrieved and difficult to justify for domain expert. However, our method provides a detailed explication of the traversed ontology classes (e.g., “Substance”), relationships (e.g., “Driving Under the Influence of Substances”), and entity instances (e.g., “Alcohol”, “Marijuana”). Moreover, we supported the findings with precise statistics from the knowledge graph database, reporting 2692 incidents for alcohol versus 2598 for marijuana. Figure 6, addressing “least used kind of drug in prescription drug misuse incidents,” further highlighted this distinction. While GraphRAG focused on specific drugs, potentially misinterpreting the query, StructuGraphRAG provided a categorical analysis, identifying “Sedative” as the least common with 146 incidents, compared to “Pain Reliever” (1523) and “Stimulant” (984).

## Discussion and Conclusion

Our evaluation demonstrates that StructuGraphRAG outperforms other LLMs-based methods like GraphRAG and GPT-4o in domain-specific knowledge representation and retrieval. The key advantage of StructuGraphRAG lies in its domain knowledge-informed ontology construction process, which leverages structured documents such as codebooks to create a more comprehensive and contextually relevant knowledge graph. This is demonstrated in both intrinsic and extrinsic evaluation. This approach not only enhances the quality of responses to complex queries but also makes it easier for domain experts to examine and follow the reasoning behind these responses. By providing a clear trail of ontological relationships and data sources, StructuGraphRAG enables experts to make more informed decisions, bridging the gap between automated systems and human expertise in specialized fields like social science research. To foster reproducibility and further impact of our work, we have made our code available on GitHub at <https://github.com/SAIL-UA/structuGraphRAG>.

In conclusion, the development of StructuGraphRAG represents a significant step forward in adapting large language models and knowledge graphs for domain-specific applications. By demonstrating improved performance in social science data analysis, our method opens up new possibilities for enhancing decision-making processes in fields where context and nuanced understanding are crucial. The broader impact of this approach extends beyond social sciences, potentially revolutionizing how we handle complex, structured information in areas such as healthcare, policy-making, and scientific research. As AI systems become increasingly integrated into specialized domains, methods like StructuGraphRAG that can effectively incorporate domain knowledge and provide transparent reasoning will be essential in building trust and ensuring the responsible use of AI in critical decision-making processes.

## References

- Besta, M.; Blach, N.; Kubicek, A.; Gerstenberger, R.; Podstawski, M.; Gianinazzi, L.; Gajda, J.; Lehmann, T.; Niewiadomski, H.; Nyczyk, P.; et al. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 17682–17690.
- Caufield, J. H.; Hegde, H.; Emonet, V.; Harris, N. L.; Joachimiak, M. P.; Matentzoglou, N.; Kim, H.; Moxon, S.; Reese, J. T.; Haendel, M. A.; et al. 2024. Structured prompt interrogation and recursive extraction of semantics (SPIRES): A method for populating knowledge bases using zero-shot learning. *Bioinformatics*, 40(3): btae104.
- Chen, Z.; Zhang, Y.; Fang, Y.; Geng, Y.; Guo, L.; Chen, X.; Li, Q.; Zhang, W.; Chen, J.; Zhu, Y.; et al. 2024. Knowledge graphs meet multi-modal learning: A comprehensive survey. *arXiv preprint arXiv:2402.05391*.
- Cuconasu, F.; Trappolini, G.; Siciliano, F.; Filice, S.; Campagnano, C.; Maarek, Y.; Tonello, N.; and Silvestri, F. 2024. The power of noise: Redefining retrieval for rag systems. In *Proceedings of the 47th International ACM SIGIR*

*Conference on Research and Development in Information Retrieval*, 719–729.

Ding, Y.; Fan, W.; Ning, L.; Wang, S.; Li, H.; Yin, D.; Chua, T.-S.; and Li, Q. 2024. A survey on rag meets llms: Towards retrieval-augmented large language models. *arXiv preprint arXiv:2405.06211*.

Edge, D.; Trinh, H.; Cheng, N.; Bradley, J.; Chao, A.; Mody, A.; Truitt, S.; and Larson, J. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*.

Feng, C.; Zhang, X.; and Fei, Z. 2023. Knowledge solver: Teaching llms to search for domain knowledge from knowledge graphs. *arXiv preprint arXiv:2309.03118*.

Gloeckle, F.; Idrissi, B. Y.; Rozière, B.; Lopez-Paz, D.; and Synnaeve, G. 2024. Better & faster large language models via multi-token prediction. *arXiv preprint arXiv:2404.19737*.

Hager, P.; Jungmann, F.; Holland, R.; Bhagat, K.; Hubrecht, I.; Knauer, M.; Vielhauer, J.; Makowski, M.; Braren, R.; Kaissis, G.; et al. 2024. Evaluation and mitigation of the limitations of large language models in clinical decision-making. *Nature medicine*, 1–10.

Harrell, B. E. 1946. The bird population in selected areas of the gorge.

Hu, Y.; and Lu, Y. 2024. Rag and rau: A survey on retrieval-augmented language model in natural language processing. *arXiv preprint arXiv:2404.19543*.

Hussien, M. M.; Melo, A. N.; Ballardini, A. L.; Maldonado, C. S.; Izquierdo, R.; and Ángel Sotelo, M. 2024. RAG-based Explainable Prediction of Road Users Behaviors for Automated Driving using Knowledge Graphs and Large Language Models. *arXiv:2405.00449*.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; Riedel, S.; and Kiela, D. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 9459–9474. Curran Associates, Inc.

Li, C. 2024. GENERAL AGENT:SELF-CALL AND STACK MEMORY. Lichenarthurdata@gmail.com.

Matentzoglou, N.; Caufield, J. H.; Hegde, H. B.; Reese, J. T.; Moxon, S.; Kim, H.; Harris, N. L.; Haendel, M. A.; and Mungall, C. J. 2023. Mappergpt: Large language models for linking and mapping entities. *arXiv preprint arXiv:2310.03666*.

Meyer, L.-P.; Stadler, C.; Frey, J.; Radtke, N.; Junghanns, K.; Meissner, R.; Dziwis, G.; Bulert, K.; and Martin, M. 2023. Llm-assisted knowledge graph engineering: Experiments with chatgpt. In *Working conference on Artificial Intelligence Development for a Resilient and Sustainable Tomorrow*, 103–115. Springer Fachmedien Wiesbaden Wiesbaden.

Pan, S.; Luo, L.; Wang, Y.; Chen, C.; Wang, J.; and Wu, X. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*.

Patil, R.; and Gudivada, V. 2024. A review of current trends, techniques, and challenges in large language models (llms). *Applied Sciences*, 14(5): 2074.

Procko, T. 2024. Graph Retrieval-Augmented Generation for Large Language Models: A Survey. *Available at SSRN*.

Safavi, T.; and Koutra, D. 2021. Relational world knowledge representation in contextual language models: A review. *arXiv preprint arXiv:2104.05837*.

Sanmartin, D. 2024. KG-RAG: Bridging the Gap Between Knowledge and Creativity. *arXiv:2405.12035*.

Sawarkar, K.; Mangal, A.; and Solanki, S. R. 2024. Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers. *arXiv:2404.07220*.

Song, Y.; Wang, T.; Cai, P.; Mondal, S. K.; and Sahoo, J. P. 2023. A comprehensive survey of few-shot learning: Evolution, applications, challenges, and opportunities. *ACM Computing Surveys*, 55(13s): 1–40.

Vekariya, D. V.; and Limbasiya, N. R. 2020. A Novel Approach for Semantic Similarity Measurement for High Quality Answer Selection in Question Answering using Deep Learning Methods. In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 518–522.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E.; Le, Q. V.; and Zhou, D. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 24824–24837. Curran Associates, Inc.

Wiratunga, N.; Abeyratne, R.; Jayawardena, L.; Martin, K.; Massie, S.; Nkisi-Orji, I.; Weerasinghe, R.; Lirer, A.; and Fleisch, B. 2024. CBR-RAG: case-based reasoning for retrieval augmented generation in LLMs for legal question answering. In *International Conference on Case-Based Reasoning*, 445–460. Springer.

Wu, S.; Fei, H.; Qu, L.; Ji, W.; and Chua, T.-S. 2024. NExT-GPT: Any-to-Any Multimodal LLM. *arXiv:2309.05519*.

Wu, Y.; Hu, N.; Qi, G.; Bi, S.; Ren, J.; Xie, A.; and Song, W. 2023. Retrieve-rewrite-answer: A kg-to-text enhanced llms framework for knowledge graph question answering. *arXiv preprint arXiv:2309.11206*.

Yao, S.; Yu, D.; Zhao, J.; Shafran, I.; Griffiths, T.; Cao, Y.; and Narasimhan, K. 2024. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36.

Zhang, N.; Yao, Y.; Tian, B.; Wang, P.; Deng, S.; Wang, M.; Xi, Z.; Mao, S.; Zhang, J.; Ni, Y.; et al. 2024. A comprehensive study of knowledge editing for large language models. *arXiv preprint arXiv:2401.01286*.