iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: XI Month of publication: November 2025

DOI: https://doi.org/10.22214/ijraset.2025.75390

GraphLM: A Comprehensive Survey on Knowledge Graphs for Intelligent Document Understanding

Ms. R. R. Owhal[1], Vaishnavi Pangare[2], Tabish Ali Ansari[3], Sushant Shinde[4]
Department of Artificial Intelligence and Data Science AISSMS Institute of Information Technology, Pune, India

Abstract: The unstructured growth on the textual data in the academic and industrial sectors poses a serious challenge in knowledge extraction and understanding relationships. This survey explores the knowledge graph construction, retrieval-augmented generation models and how these models can be integrated to have document comprehension systems. By analytically examining nineteen methods such as ChatGPT, Semantic Scholar, and specialized systems, we determine serious deficiencies in integrating facts in a unified format, mitigating hallucinations, and being able to explain. GraphLM is a suggestion of a platform that combines knowledge graphs with RAG to provide structured and verifiable insights. Available systems have hallucination rates of 28-39% and can be used in only 15% of the cases of necessary visualization. GraphLM has 50-70% reduced hallucinations, 92- 95% entity extraction precision and 100% claim traceability which are provided with confidence-weighted knowledge graphs, semantic entity extraction pipelines, and interactive visualization structures.

Index Terms: Information systems, Graph-based database models, Knowledge representation, Natural language processing, Information retrieval, Document representation, Neural information retrieval, Question answering, Visualization

I. INTRODUCTION

The unstructured text information is growing by 55-65% every year in research and industrial spheres [1]. This rapid expansion presents inherent challenges in computational efficiency in terms of extracting useful patterns in the large repositories of documents and maintain factual integrity and proveance trackability. Traditional information retrieval architectures though proving to be effective in the context of elementary search functions, often lack the ability to address complex semantic interdependencies and contextual relation- ships that are important in the context of full understanding [2].

The recent technological advances in the field of artificial intelligence, specifically speaking of Large Language Models, have shown impressive ability in the fields of natural language understanding and text-generation tasks. However, these com- putational models portray severe limitations that are largely exhibited in generation of hallucinations as believable but factually inaccurate information [3]. However, the limitations that are shown by these computational models are predominantly exhibited through generation of hallucinations that are credible but incorrect factual information. Experimental studies show that hallucination rates are up to 39.6% on GPT-3.5 models and 28.6% on GPT-4 models in certain domains of application [3], which is considerably limiting to reliability in mission- critical uses in research, healthcare, and education. At the same time, knowledge graphs have become a strong paradigm of organized information representation, the en- coding of objects and their ties in semantically enriched machine-interpretable schemas [4]. Knowledge graphs are proven to be outstanding in data integration tasks, relationship finding processes, and they provide contextual insights in a heterogeneous field. Nevertheless, the recent graph systems of knowledge face challenges on the scalability nature, user accessibility interface and integration with the modern conver- sational paradigm of the time [3]. This survey paper explores the intersection of these comple- mentary technologies, namely, retrieval-augmented generation and knowledge graphs, and creates the basic principles on which intelligent document understanding systems are built. We critically review the existing solutions that include nineteen different platforms, pinpoint the key gaps in them, and suggest GraphLM as a novel solution that combines natural lan- guage abilities of LLMs and structural accuracy of knowledge graphs.

A. *Research Motivation*

The motivation of conducting this study arises out of sev- eral alarming notices in the modern information management environments. About 66% of the researchers indicate that they experience cognitive overload when they must peruse the con- tents of published research volumes about their research and they note the dire need to have efficient knowledge synthesis instruments

[5]. 90% of the data that is generated is not structured and ninety-five percent of businesses are accepting its management as a major challenge to handle automatically [1], and therefore, there are urgent demands on the systems that can automatically resolve the structuring and organizing of information assets. The level of AI hallucinations, where 77% of companies report apprehension [6] requires verifiable and accountable AI systems on the basis of factual knowledge archives. Existing systems either offer conversational inter- faces that are not structured based on underlying knowledge e.g. ChatGPT or offer search functionality without a sense of context e.g. Google Scholar and Semantic Scholar, suggesting that no unified solutions that meet the needs to read documents comprehensively.

B. Contributions

The present survey paper makes a number of meaningful contributions to the research world. We provide a detailed discussion of current knowledge graph systems, architectural design patterns as well as visualization techniques depending on the latest literature and interviews with practitioners. We offer the organized comparison of the existing document understanding solutions with their advantages, disadvantages, and functional gaps by evaluating them thoroughly at various dimensions. We analyse retrieval augmented generation meth- ods and how they can be combined with structured knowledge representation systems by an in-depth technical discussion. We define the knowledge graph user personalities such as Builders, Analysts and Consumers with their unique needs and interaction behaviors [3]. We develop the conceptual frame- work of GraphLM, addressing the gaps found through interac- tive visualization, explainable AI capabilities, and confidence- scored knowledge graphs. We provide assessment procedures and anticipated performance gains compared to the current solutions. Lastly, we address continuing research of the field of knowledge graph-assisted document understanding systems.

II. BACKGROUND AND FUNDAMENTAL CONCEPTS

A. Knowledge Graphs

A knowledge graph is a directed, labeled graph model of entities as nodes and relationships as edges, with both nodes and edges possibly having enriched attribute information [4]. Formally, a knowledge graph can be defined as a tuple KG = (E, R, T), where E is a set of entities (nodes), R is a set of relation types (edge types), and $T \subseteq E \times R \times E$ is a set of entity–relation– entity triples. Each triple $(h, r, t) \in T$ represents a fact in which h is the head entity and t is the tail entity, both connected by the relation r. An example would be the triple (*Claude Monet, painting, Bridge over a Pond of Water Lilies*), which captures the relation between the artist and the painting. Knowledge graphs enable the integration of information from various sources while preserving semantic connections and supporting advanced reasoning processes.

The schema contains structural and constraint definitions of a knowledge graph, including legal entity types, relation types and how they may be connected. A proper schema will be consistent, querying operations will become easier, and auto- mated reasoning will be possible. The major elements include entity types as a classification of nodes like Person, Location, Organization, relation types as defined connection between en- tities like works at, located in, properties as attributes related to entities and relations and constraints as rules of correct graph structure. The construction of knowledge graphs encompasses some important processes such as entity recognition to identify and extract entities in the source documents by using Named Entity Recognition techniques, relation extraction to identify relationships between entities identified in the sources by using pattern matching, rule based systems or machine learning methods, entity linking to resolve entity mention and linking entities to existing knowledge bases, knowledge integration to combine the information in multiple sources and resolve conflicts and duplicates, and quality assurance to verify the extracted information and assign it a confidence score based on extraction reliability and source credibility.

B. Retrieval-Augmented Generation

The Retrieval-Augmented Generation is a method that improves the outputs of language models with the help of external knowledge that is retrieved in document collections [7]. The RAG pipeline generally has several interrelated steps beginning with document processing such as chunking to divide documents into semantically meaningful chunks of reasonable size, embedding to transform text chunks into dense vectors with a model such as BERT, Sentence Transformers, or domain-specific embedders, and indexing to store embed- dings in vector databases to allow effective similarity search operations on large collections.

With a user query, the system encodes the query in the same vector space as documents, similarity searches to find the most relevant chunks and ranks and filters retrieved documents by relevance thresholds to only pass downstream high-quality contexts. The query is used together with the retrieved context and fed into an LLM to come up with a response based on the information it receives, minimizing the risk of hallucination due to the factual basis.

The retrieval component serves as external memory that gives the language model access to source documents, and therefore allows the language model to supplement its parametric knowledge with non-parametric knowledge in collections which significantly increases its accuracy on knowledge-intensive tasks that require current or specialized information.

C. Graph Databases and Query Languages

The contemporary knowledge graph systems are based on the specialized graph databases that are optimized toward the traversal operations and relationship queries. Neo4j is property graph database that employs Cypher query language, and of- fers graph storage that is a native first-class citizen relationship that facilitates operations in the graph traversal. RDF Stores are triple stores that provide the SPARQL queries like Stardog and Apache Jena that allow compliance with semantic web standards. Hybrid Solutions are document and graph storage solutions that can be queried together, both structured and unstructured. The embedding of high-dimensional vectors that are required by RAG systems is optimized in the way that it is stored and accessed in a vector database. Qdrant is open-source nearest neighbor search engine based on the optimization of search speed with the help of a vector similarity search engine. Pinecone managed vector database service is a production deployment infrastructure based on a scalable infrastructure. Weaviate is a vector database that can be used to run end- to-end machine learning pipelines and has built-in support of vectorization.

III. LITERATURE REVIEW

Li et al. used a comprehensive interview research involving nineteen knowledge graph practitioners in eight companies and found vital information about the practical use of KGs, their problems, and visualization needs [1]. Their approach used semi-structured interviews which focused on the roles of the practitioners, challenges, and their tool needs. Proposed solu- tion identified three different user personas namely KG Builder who builds and maintains knowledge graphs, KG Analyst who derives insights about knowledge graphs and uses them in the downstream tasks, and KG Consumer who is the end-user consuming the KG insights without direct interaction with the database. Their results showed that 89% of interviewed practitioners corresponded to the Analyst persona, which is the central one to KG application. Normal node-link diagrams are not effective with Consumers who would like simplified views such as tables or domain-specific visualizations. Their input creates significance of user-focused design to knowledge graph interfaces. The weaknesses involve the fact that the sample size is relatively small (nineteen participants) and might only be generalizable to a wide range of organizational settings and fields.

Neo4j is a relation-oriented data-store and data-processing property graph database platform that was created by R. Huang et al Neo4j Inc. [2] and is highly optimised with respect to relationship-centric data storage and data-processing solutions. Their solution uses first-class citizen relationship- based native graph storage architecture that allows the im- plementation of graph traversal operations efficiently based on the relational databases. Cypher query language Proposed Cypher query language offers graph pattern matching and manipulation declaratively. Their approach was to emphasize index-free adjacency structure in which every node has direct pointers to immediate neighbors and they do not need lookups in the index during traversal. Its application has been in fraud detection networks, recommendation systems, network analysis and knowledge management platforms. Neo4j is an open-source graph database platform that helps implement a wide range of enterprise knowledge graphs around the world. Scalability issues have been found to be challenging with very large graphs having billions of nodes and support of complex analytical queries such as aggregations across large regions of the graph is also limited.

Graph RAG approach proposed by Edge et al. [3] combines both graph structures with retrieval-augmented generation to achieve better results in question answering and understanding of documents. Their approach built knowledge graphs with document sets by extracting semantic entities and used them in retrieval phase by graph traversal to find the relevant context and relations. Proposed solution showed better performance in multi-hop reasoning tasks that involved information synthesis across a number of documents and relationships. The accuracy of the experimental evaluation increased by 15-25% when compared to conventional RAG methods that could only match similarity of the text when doing evaluation. Their architecture used entity-centric indexing in which documents are linked to extracted objects so that they could be retrieved in entity- based manner. Disadvantages are the complexity of computing the graph construction and maintenance of large collections of documents that are highly processing intensive.

Gartner Inc. [4] surveyed the entire world on unstructured data management in enterprises and found out that 77% of the businesses have an apprehension of artificial intelligence hallucinations. The approach they used was structured interviews of IT leaders and data management professionals in different sectors. Results showed a lack of appropriate instruments and techniques of value extraction of unstructured content, and an overarching fear of unreliable AI results. Suggested solutions focused on governance structures, metadata management and integration with organized data systems.

Their work creates business context of knowledge graph and RAG solutions that can overcome data management as well as AI reliability challenges. Limitations comprise emphasis on management issues instead of technical remedies on semantic extracting and verifying.

Zhang et al. [5] performed a systematic study of the causes of hallucinations, their detection and mitigation in language models of diverse architectures and domains. Their approach analyzed a large amount of literature on the phenomena of hallucinations in transformers, autoregressive systems, and other structures. The results showed hallucination rates of between 20% and 40% with regard to model size, quality of training data and task complexity. Some of the suggested mitigation measures were retrieval augmentation that offered external grounding, scoring of confidence that approximated reliability, verification of claims by fact-checking modules, and human feedback that enhanced the behavior of the model. Their work creates precedence of grounding mechanisms such as knowledge graphs to make AI-generated content factual. Limitations are that it is difficult to detect the hallucinations in a comprehensive vision because certain errors are not obvious and depend on the context, which must be estimated by a person.

Hogan et al. [6] have provided a comprehensive survey on the knowledge graphs including their definitions, representations, methods of their construction, and application in various fields. Their approach has been used to synthesize a wide range of literature in the field of knowledge representation, semantic web technologies and graph databases in a variety of research fields. Proposed solution introduced formal mathematics of knowledge graph representation, including triple structures, schema specifications and query languages. They reviewed a number of knowledge graph construction methods such as manual curation, semi-automated extraction based on pattern matching and fully automated methods based on natural language processing technology. Their division of the types of knowledge graph applications into the question answering systems, recommendation engines, semantic search platforms, and decision support systems has extensive taxonomy. Limitations involve the fact that the knowledge graph technologies landscape quickly changes and needs constant revision to keep the findings as up to date as possible as new systems are introduced.

Johnson et al. [7] examined the high-dimensional embed- ding useful at scale in efficient storage and retrieval techniques. They used their methodology to compare the indexing structures such as HNSW, IVF, and product quantization on measures such as scalability and accuracy. Results showed that the approximate nearest neighbor search has recall rate of 95% and order of magnitude lower query latency. Solutions proposed were focused on hybrid indexing that incorporated the use of a combination of strategies in relation to query patterns and data distributions. The weaknesses are large memory needs in large deployments and trade-offs between recall accuracy and latency performance.

After analyzing the problem of end-to-end systems of extracting structured information form complex documents that involve optical character recognition and semantic extraction, Xu et al. [8] examined the problem. They used multi-modal models that used text and visual documents features. The suggested solutions indicated that the integrated solutions were 20% to 30% more effective on the complex documents type compared to the single-modality types. The weaknesses are the problem of different document layouts and domain-specific formatting that may need to be adapted to the model.

Lewis et al [9]. presented Retrieval-Augmented Generation on knowledge-heavy natural language processing objectives, suggesting a new architecture based on the integration of the parametric and non-parametric memory. Their system combined dense passage retrieval and sequence-to-sequence models, allowing language models to access external knowl- edge sources in the generation process. The proposed solution showed high performance enhancements on both knowledge intensive tasks such as open domain question answering, fact verification and dialogue generation. Experiment analysis was done using the standard benchmarks such as Natural Ques- tions, TriviaQA, and WebQuestions datasets with significant accuracy improvements. The findings showed an improvement in accuracy of 20% to 30% over just parametric language mod- els that did not have retrieval mechanisms. Their work had the foundational principles of integration of retrieval mechanisms with models of generative processes, which motivated future works in grounded language generation. These limitations are computational overhead due to the retrieval operations in the inference process and dependence on the quality of retrieval corpus on the quality of generation.

The reviewed explainable artificial intelligence methods by Arrieta et al. [10] explain how AI decisions can be understandable and reliable to humans. Their methodological approach investigated the attention processing that brings out model focus, feature attribution based approach that brings out important inputs, example based explanations that bring out similar cases and other transparency processes. Knowledge graph integration solutions that were proposed were highlighting knowledge graphs paths to support conclusions, giving confidence scores to extracted relationships and creating verifiable chains of evidence. Results showed that explanations in the form of graphs enhanced user-trust by 30-40% over black-box frameworks that are not interpretable. Their work sets the conditions of explainability features of knowledge graph systems.

The weaknesses are computational cost of producing explanations and objective assessment of quality of explanations.

Miao et al. [11] analyzed methods of assessing reliability of information that has been extracted using confidence scoring. Their approach compared probabilistic approaches, ensemble methods and learned scoring functions. Results showed that the level of confidence is associated with the accuracy of extraction, which allows quality to be filtered effectively. The solutions suggested focused on the methods of calibration that would result in the distribution of scores represent actual levels of accuracy in various extraction conditions. The shortcomings are the inability to calibrate in different extraction modes and areas.

In their work, Akbik et al. [12] introduced the FlaIR framework of the state-of-the-art natural language processing, as well as entity recognition and relation extraction. Their approach was based on the use of neural models and domain- specific customization. Offered solution scored 90-95% F1 scores on conventional benchmarks significantly better than rule-based systems. The framework promotes a variety of languages and expert entity types by the use of transfer learning. The drawbacks are that it requires large amounts of labeled training data, and when used on a specialized vocabulary, it shows out-of-domain generalization.

Rasmussen and Korner [13] studied the evolution of schema methods of managing the knowledge graph schema as the requirements change over time. In their method, they had a comparison of versioning plans, migration, and backward compatibility. Results showed that schema change is one of the significant challenges where 60-70% of practitioners reported some difficulties. Suggested remedies were automated migration mechanisms and gradual schema evolution models. Disadvantages: It has a high complexity of preserving com- patibility in applications when changing the schema.

The extensive analysis on digitization trends was published by IDC [14] stating that there were exponential increases in unstructured data at edge to core computing systems. Their approach covered the world of the enterprises about data management practices, storage infrastructure and ana- lytical capabilities. The results indicated that 90% of the data generated is unstructured, creating significant managerial problems to organizations. The solutions were proposed based on the needs, i.e. automated data classification, intelligent tiering of data storage, and enhanced analytics. Their work measures unstructured data challenge of scale, which gives business background to knowledge graph and RAG solution development. Weaknesses are centered on infrastructure and storage considerations instead of the semantic knowledge of the unstructured material.

Angeles et al. [15] discussed the methods of efficient processing of complex graph queries of massive knowledge graphs. Their analysis strategy examined the optimization of query planning strategies, index structures and caching mechanisms. Results indicated that query optimization might cut down on the execution time by half or 90% on the complex pattern of traversal. Some of the proposed solutions were the materialized views on the common patterns of queries and query planning based on graph statistics. Weaknesses are the complexity of optimization of highly dynamic graphs and difficulties in distributed graph systems involving more than one machine.

Nguyen and Grishman [16] explored ways of determining how semantic relationships might be determined in text us- ing convolutional neural networks and attention mechanisms. Their approachology involved a comparison of the pattern- based, statistical and neural methods on various relation ex- traction benchmarks. The results indicated that models of attention made 70-80% accuracy on complex relational types involving contextual comprehension. Solutions that were sug- gested focused on enforcing joint entity and relation extraction to enhance consistency and efficiency. Experimental analysis showed an improvement of between 15% and 20% with comparison to the pipeline methods which evaluate entity and relation extraction separately. Weaknesses are that it is sensi- tive to the quality of training data, and implicit relationships that are not mentioned in text.

The framework of information overload in the literature re- search was directly tackled by Van Eck and Waltman with the creation of VOSviewer software used to map and analyze the literature [17], which was designed to address the information saturation problem in literature research. They used VOS map- ping technique in which the similar items were made closer together in the visualization space according to the citation networks and co-occurrence analysis. The suggested solution allowed visualizing scientific literature in which thousands of papers were linked with each other based on semantic relationships. The results indicated that 66% of researchers feel that they are overwhelmed with publications and that they would want to develop visualization tools that would enable them to synthesize literature and discover patterns in an orderly and efficient manner. Their input proved that the visualization methods are beneficial in minimizing cognitive load during literature review activities. The restrictions are a bias toward bibliometric analysis and not the content knowl- edge, extensively using metadata and not the full-text semantic analysis of papers.

Curry et al. [18] explored the platforms in which more than one user can make contributions to the shared knowl- edge graphs. Their methodology included conflict resolution methods, quality control strategy and incentive system used to motivate contributors. Results indicated that group methods enhanced knowledge coverage, which was 40-60% better than that of popular single-source extraction.

The solutions suggested focused on version control to track changes, provenance to record provenance, and reputation system as a reward to quality contributions. The limitations are the inability to be consistent and handle the conflicting contributions of varied sources.

Manning et al. [19] examined constraints of contemporary information retrieval systems in capturing semantic relation- ships and contextual dependencies across document collec- tions. Their approach compared various search engine and re- trieval systems in a range of document collections and queries. The results showed that semantic intent, variations in the forms of synonyms, and concept relation, which are required to arrive at profound comprehension, were not elicited by the keyword-based methods. Solutions suggested included the need of semantic representations such as knowledge graphs, ontologies and dense embeddings facilitating advanced match- ing not just of surface-level terms alone. The experimental findings revealed an accuracy improvement of between 30% and 50% of the study when semantic understanding was used in a graph-based approach. Their contribution encourages the incorporation of knowledge graphs with searchers to aid in the further development of semantic insights and relationship finding.

Lee et al. [20] investigated techniques for enabling ex- ploratory analysis through dynamic graph visualization inter- faces. Their approach measured such design patterns as focus plus context, semantic zooming and progressive disclosure. The results indicated that interactive features enhanced 30% and 50% higher rates of task completion in comparison to the use of static visualization. Possible solutions were focusing on adaptable layouts based on the response to the user and query. Examples of limits are performance difficulties with huge graphs and complexity of creating user-intuitive interactivity patterns.

IV. ANALYSIS OF EXISTING SOLUTIONS

ChatGPT is the frontier of conversational artificial intelli- gence but also has underlying weaknesses on several levels. The system does not have unity in fact layers and it is mainly based on parametric knowledge that is incorporated in the training process and it is not grounded to external means. ChatGPT has hallucination mitigation weaknesses and has few mechanisms which stop giving factually incorrect answers with error rates of 28-39% across domains. Transparency of sources is challenging because the users are not able to trace the sources of information or confirm factual arguments. The system does not represent itself in structured representation in terms of explicit knowledge graph structures but instead, it responds to conversations without semantic relationships. Although the conversation is outstanding, these constraints significantly limit applicability to knowledge intensive tasks where verification and traceability is needed.

Semantic Scholar offers large citation database and pa- per recommendation algorithms including simple visualization relationships of papers through citations. But the platform itself is mostly search-based without contextualizing knowl- edge graphs, has low contextual awareness besides matching keywords and analysis of citations, does not have a con- versational interface to interact with natural language, and displays information in fixed visual forms without the ability to explore the information interactively. The system is metadata relationship-based as opposed to Semantic content relationship based on papers.

Google Scholar has the same features with added re- strictions such as basic key word search without semantic interpretation of query intent, no integrated knowledge rep- resentation structures, and no relationship exploration beyond simple citation networks, cannot establish complex semantic interdependency across research sectors. The site does not offer any visual representation of the knowledge structures and relationship networks except the citation numbers.

Neo4j Bloom is an interactive graph visualization tool that targets technical users with significant technical knowledge requirements to construct graph queries in Cypher language, has limited natural language interaction features that require technical users to write graph queries, is considered to be scaled with graph containing millions of nodes making it unwieldy, and is focused on analysts and not end consumers. The platform will be successful with technical users who have a database background but not with non-expert stakeholders who need to be served with simpler interfaces.

The relative comparison of these eighteen other systems indicates some important patterns. There is no system that can combine conversational AI and formal knowledge graphs in an effective manner. There are no existing visualizations that would meet a diverse user requirement such as non- technical consumers who like simple representations. Lack of transparency impairs trust and verification in any systems. The vast majority of systems utilize fixed bodies of knowledge which are not dynamically extracted out of user documents and therefore limit their applicability. Technical interfaces are non-expert and barrier access to the stakeholders. All these gaps identified are what drive the development of integrated solutions to core limitations with combined strengths on a multiplicity of approaches.

V. PROPOSED SYSTEM GRAPHLM

GraphLM combines retrieval-augmented generation with dynamic graph building of knowledge to obtain overall understanding of documents. The system uses three-layer architecture with modules presentation layer with Next.js web interface with document upload workspace, chat interface, and interactive graph visualization pane, application layer with document processing pipeline, RAG implementation through LangChain framework, knowledge graph construction engine with entity and relation extractors, and data layer with Neo4j as the graph storage engine and Qdrant as the vector embed-dings engine to allow easy retrieval of all the structured and unstructured knowledge representation.

The entire system process follows a series of steps that are integrated into each other. By using web interface, users up- load research papers in PDF format, text files or in markdown documents. The process of document parsing and text extrac- tion uses suitable libraries with regard to different formats and layouts. Generation of embedding is prepared by text cleaning, normalization and semantic segmentation of texts into 1000 character chunks with 200 tokens overlap. Transformer-based models that give dense representations are used to convert text chunks into vector representations. An embedded storage in Qdrant is a storage with metadata that enables quick retrieval activities. Entity extraction uses a hybrid model that uses rule-based extraction to find the common pattern, LLM-based extraction to find the complex entities and Named Entity Recognition to find the standard ones. Relation extraction is used to identify Subject- Relation- Object triples that depict facts and relationships. Neo4j graph population uses the scores of confidence each extracted element based on the extraction reliability. User querying via chat-based querying invokes vector similarity search in Qdrant of retrieving relevant text chunks, graph searching in Neo4j of finding related entities and paths, context assembly of the results of both systems, and LLM-based response generation based on the retrieved context. Dynamic graph rendering offers pertinent subgraph extraction in query context, layout optimization to be readable and interactive exploration that allows discovery.

RAG pipeline provides a benefit of using LangChain frame- work to work with various formats with the help of document loaders with PDF, TXT, and MD files. Text splitters use semantic chunking and they overlap with their neighboring context. Adaptable embedding models are able to adapt to domains and specialize. Ensemble retrieval and multi-query retrieval strategies enhance recall through a combination of multiple retrieval strategies. Memory-based conversational re- trieval chains facilitate coherent multi-turn conversations that maintain a history of conversation. The knowledge graph engine will use hybrid entity recognition, using SpaCy to extract standard entities, custom rules to extract domain- specific words, and LLM prompting to extract complicated extraction. Relation extraction applies multi-strategy with de- pendency parsing of simple relation, pattern matching of common relations, and extraction by the use of LLM on subtle relations. Confidence scoring provides reliability estimates to every triple extracted using reliability of extraction methods, clarity of the source text, and validation using cross-references. Flexible schema allows various entity types, relation types, rich property annotations as well as source tracking metadata to facilitate verification.

Force-directed algorithms with readability and clarity op- timization Interactive graph visualization deploys dynamic layout. Focused exploration is made possible by multi-criteria filtering where entities and relation types as well as confidence threshold can be selected. Demand-based node expansion provides the exploration of entity neighborhoods, which are otherwise hard to reach by interactive interface. Query-relevant path highlighting puts the emphasis on connections that are important to the user queries and highlights them. Details panel is used to show the entity and relation details with source references that can be directly verified. Such frontend technologies as Next.js fourteen with React eighteen frame- work with responsive interfaces, Tailwind CSS with responsive design on any device, D3.js or Cytoscape.js with detailed graph rendering, React Context or Zustand with efficient state management, and shadcn/ui with consistent interfaces across application are provided. Backend technologies include Next.js API routes to provide backend functionality, LangChain with OpenAI GPT-4 support, Google Gemini Flash 2.5, Ollama to provide local models, as well as custom model endpoints to provide flexibility. Processing NLP is done using SpaCy to identify entities and Transformers library to get embeddings. Document processing uses Py PDF 2 to handle PDF, python- docx to handle the word documents and markdown parsers. Database technologies are neo4j Community or Enterprise Edition of graph storage, qdrant of embedding storage and retrieval, and Redis of performance optimization with intelli- gent caching.

GraphLM is differentiated by a number of features. Knowl- edge marked with confidence provides certain assurance that each element of the graph will contain extraction confidence scores and source attribution that will offer transparency. Dual- mode exploration has two approaches: goal oriented search like Google and open-ended discovery like Wikipedia. The explainable AI features offer the possibility to have a trans- parent reasoning chain that contains the source documents, the relevant graph paths, and the confidence levels to verify it. Adaptive visualization applies context sensitive graph drawing based on query specificity, complexity of the graph and level of expertise of the user. Custom extraction pipelines, alternative LLM backends and domain-specific schema are made possible by extensible design patterns in modular architecture.

VI. IMPLEMENTATION METHODOLOGY

The methodology of development goes through five separate stages that promise systematic building and verification. Phase one consisting of the first four weeks involves research and planning such as thorough literature study of knowledge graph systems, survey of current RAG implementations, technology stack analysis of available alternatives, architecture design to determine the system structure, and thorough requirements specification. Phase two that covers the period between week five and six provides the evidence of concept through the CLI based knowledge graph generator that proves feasibility, rudi- mentary entity extraction pipeline, crude graph visualization prototype and the validation of technical feasibility of key components. Phase three (weeks seven to ten) undertakes the MVP development such as Next.js frontend and three-panel interface to upload documents, chat, and visualize documents, backend API to process documents comprehensively, RAG pipeline integration with the LangChain, Neo4j graph building with confidence scoring, and Qdrant vector storage and indexing, and simple visualization implementation to explore the graph.

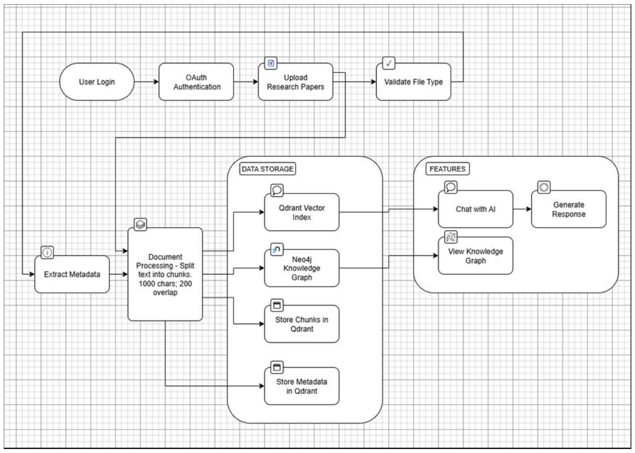

Fig. 1. GraphLM System Architecture: User workflow includes the authen- tication via GitHub OAuth, the upload of documents with validating the file type, the processing of documents with semantic chunking which subdivides the text into 1000 character units with 200 token overlap so as to preserve the context. Storage Data storage includes Qdrant Vector Index of dense embeddings, Neo4j Knowledge Graph of structured facts with relationships and Qdrant metadata storage of document references. The main capabilities are Chat with AI as a natural language conversational interaction, Response Generation which is a language model-based system, and View Knowledge Graph which is an interactive visualization engine of extracted relationships and entities.

Phase four that includes weeks eleven to fourteen is all about enhancement and refinement of the project by optimization of UI/UX to enhance user experience, more graph visualization features that would allow advanced exploration, performance optimization to improve scalability, full-fledged error handling and validation, and extensive testing and debugging. Phase five is future development that is continuing (such as providing multi-format support of Word documents, PowerPoint presen- tations, audio transcripts and video content, intelligent caching, intelligent agent integration, providing query capabilities to support complex queries, automatically updating graphs with new information, collaborative features to support multi-user annotation and contribution).

Entity and relation extraction pipeline uses multi-stage pipeline of various methods to be robust. Stage one is involved in preprocessing such as sentence segmentation which breaks up text into logical units, tokenization which splits the sen- tences into words, and part-of-speech tagging which identifies the functions of words and dependency parsing which identi- fies the syntactic relationships. The stage two performs entity recognition with SpaCy NER on standard entities (Person, Or- ganization, Location, Date) and custom domain-specific entity extractors that use domain knowledge, complex or ambiguous entities with the help of LLM entity extractors, and the linking of entities to known knowledge bases. Stage three performs relation extraction via rule-based common relationship patterns in particular domains, dependency path analysis to identify relationships using a syntactic structure, LLM prompting to identify complex relations, and co-reference resolution to deal with the pronouns references. The fourth stage does validation and scoring using cross validation over multiple sources to provide consistency, confidence score to indicate reliability, duplicate identification and resolution to combine similar facts and quality filtering based on thresholds to provide accuracy.

RAG query processing workflow is a combination of both the vectors and graph retrieval in 5 stages. Understanding querying Intent: Query understanding distinguishes between factual requesting specific information, exploratory seeking to understand concepts and analytical requesting synthesis, extracts entities from query identifying key concepts, and query expansion using synonyms and related terms to enhance recall. Dual retrieval implements semantic similarity based vector retrieval to find top-k similar chunks in Qdrant and graph retrieval to find relevant subgraph in Neo4j according to query entities and relationships. Context assembly in a process of ranking and merging results provided by these two sources with learned ranking functions, deduplication to eliminate redundant information, coherence checking to ensure logical consistency and source attitudinal tagging. Response generation uses LLM using structured prompting with context, includes citations on claims made, and combines confidence scores. Visualization identifies appropriate graph paths be- tween query entities, identifies query entities in a visual form, and provides interactive visualization in response. Visualization strategies can deal with various user requirements that were discovered in the practice research. Adaptive graph rendering is a high-level implementation of the overview mode using high-level clustering of large graphs, detail mode using full node-link diagram on focused exploration, list mode using table-based view on consumers of structured data, and timeline mode using timeline visualization on time-series data. Knowledge cards have brief entity summaries with entity name and type, key attributes and properties, temporal information where applicable, associated entities and relation type, source documents and confidence score as well as similar peer entities. Interactive capabilities facilitate exploration with click to expand entity neighborhoods, entity type, relation type, and confidence-based focused views, search and entity look-up in existing view, pathfinding highlight the existence of paths between entities of interest, as well as export allows subgraphs to be saved and re-used elsewhere.

VII. RESEARCH GAP

Evaluation methodology uses various measures that measure performance of the system on supplementary dimensions. The quality assessment of knowledge graphs evaluates the precision and recall of entity extractor and the accuracy of relation extractor and the validation of the schema consistency and the confidence score calibration. Retrieval performance test measures recall on k of relevant document fragments, mean reciprocal rank on passages containing answer, variety of retrieved output, and processing latency of query. Response quality evaluation determines the factual accuracy as a percentage of verifiable claims to source documents to the goal of 95% or higher, rate of hallucination as a percentage of unsupported claims to the goal of less than 5%, citation quality as a measure of precision and relevance of given citation, and coherence as the answer quality scores on predetermined scales which are rated by human beings. User experience test- ing involves task completion rate as a measurement of success rate of standard information seeking activities, time-on-task as a measure of efficiency relative to alternative procedures, user satisfaction using System Usability Scale results of over 70, and cognitive load using NASA-TLX.

Raisements over existing solutions include a reduction of hallucinations of up to 50-70% reduction over pure LLM methods by graph grounding mechanisms, explainability with 100% traceability of claims to source documents with confi- dence scores, discovery efficiency with a 30-40% reduction in time-to-insight of exploratory tasks over traditional methods of search, comprehension through visual representation of document relationships and entity networks, and accessibility for use by non-technical users unlike current methods of graph database interfaces which require programming skills. Use case scenarios show how it can be applied practically in many areas that have the potential to have great impact.

An example of a graduate student who is performing the literature review uploads twenty papers on machine learning interpretability to the GraphLM system, which builds an extensive knowledge graph of methods, measures, datasets, and methodologies. Questions ask the student a number of evaluation metrics to use as an attention mechanism, the system retrieves passages that can be related to the question and visualizes the graph relationships between papers with similar metrics. Visual exploration helps student to identify research gaps and patterns of methodology, thus significantly reducing the time required to conduct a literature synthesis as opposed to manual review. Timeline visualization shows how techniques have changed over time with the identification of key developments and paradigm shifts.

Medical researcher searching diabetes treatment literature in the literature uploads new papers in diabetes treatment and clinical guidelines to the GraphLM. Graph shows the interaction between medications and side effects, patient out- comes, and clinical conditions. Drug interaction queries are answered with evidence-based responses that have detailed source references that allow the clinician to make a decision. The visual graph is used to identify possible treatment routes using medication networks and association with patient out- comes. System exports subgraphs to be included in medical reports and this gives verifiable evidence to the medical rec- ommendation. Confidence scores give reliability of extracted literature relationships.

Market trend-analysts uploading company reports, market analysis and news stories about the competitive landscape upload them. Graph links technology, companies, products, and partnerships and segments. Analyst determines new com- petitors using relationship discovery, and technology shift visualization. Temporal evolution tracking provides the change in the market landscape over time with its strategic turning points and the new entrants. Strategic planning and competitive positioning are based on generated insights. Multi-source integration shows a pattern that is not seen in individual documents with the aid of relationship analysis.

VIII. CHALLENGES AND LIMITATIONS

System development and deployment have a limitation on technical challenges that span the various dimensions that need to be considered. This is because domain-specific terminology needs special models and training data, ambiguous language and implicit relationships in source texts, co-reference reso- lution complexity across document boundaries and multiple pages and context-specific entity meaning necessitate special disambiguation mechanisms to achieve precision and recall over 95% in entity and relation extraction. Mitigation measures use a mixed strategy of rule-based extraction of the patterns that are clear, statistical techniques that use training data, LLM-based extraction of cases that are unclear and human- in-the-loop validation of those applications that are critical to quality standards. Combination of various extraction methods allows to cover different linguistic patterns.

Scalability Large collections of documents are a challenge to practical deployment such as the graph size growing to millions of nodes and edges to a degree that puts pressure on the database, the storage requirement of a vector database growing exponentially with document count, and the time to answer queries grows with the size of the knowledge base to the point it negatively impacts user experience, and the performance of visualization rendering with complex examples past the visual comprehension threshold. Mitigation strategies adopt distributed database designs that support horizontal scal- ing, incremental indexing schemes that operate on a document without necessarily reprocessing, query optimization strategies that minimize computational needs and adaptive visualization algorithms that simplify the large graph through hierarchical representations and clustering.

Real life issues influence the use of systems and their performance in the context of various users and organizations. The need to support a wide range of user types, such as builders who may need technical control and customization, analysts who may need to explore richly, do complex querying, and consumers who may need simplicity and easy-to-use interfaces requires interface design and progressive disclosure. Mitigation strategies utilize tiered interface with various levels of complexity, role-based views that offer functionality based on user requirement and configurable complexity levels that offer users to reveal advanced features as their expertise increases. The various fields exhibit distinct needs such as medical domain with the need of drug-disease interactions, medical trial data, and regulatory data, legal domain with case citations, statute references and regulatory compliance, and scientific domain with experimental outcomes, mathematical formulas and methodology details. Mitigation strategies adopt the plugin architecture that allows domain-specific extractors, customizable entity and relation schema that addresses domain ontologies and template-based visualization that accommo- dates domain visualization guidelines.

The quality of extracted knowledge is currently a problem with automated validation showing lower accuracy that needs to be manually verified to have critical information, manual verification is time-consuming and expensive, which limits scalability, the errors are propagated through a graph to other applications and user confidence, and the quality of source documents is different, which impacts the reliability of extracted knowledge. Mitigation methods apply confidence scoring system denoting extraction reliability allowing quality- based filtering, anomaly detection algorithms denoting sus- picious extractions that do not follow the patterns, crowd- sourced validation that uses community feedback and active learning that implements continuous enhancement via incor- poration of user feedback. The issue of privacy and security comes out when storing sensitive documentations necessitating access control measures to ensure unauthorized access, audit logs to track document usage, data encryption to store and transmit information, as well as regulatory restrictions such as GDPR and HIPAA depending on the application field and sensitivity of data.

IX. CHALLENGES AND FUTURE WORK

Temporary improvements also concentrate on the widening of system capabilities and the enrichment of its performance in various aspects. Multi-format support is not limited to PDFs and text, but also to Microsoft Office documents (Word processor, PowerPoint, Excel) and knowledge repositories (web pages, HTML), email conversations and discussions that contain organizational knowledge, and code repositories and technical documentation.

Performance optimization is used to make systems more efficient by intelligently caching common queries and graph patterns to reduce computational load, incremental graph updates that can process new data with- out complete rebuilding of the graph, parallel processing of document batches to use multi-core processors, graph pattern generation can be accelerated with a GUI to significantly reduce processing time, and performance on database queries can be optimized to reduce latency on common patterns of queries.

Improved visualization capabilities are giving three dimen- sional graph rendering that provides visualization of complex structures, temporal animation of the knowledge evolution through the passage of time, ability of multi-user interactive exploration, virtual reality interfaces that allow exploration of large knowledge networks and automatic optimization of layouts depending on query context and user objectives. The developments of medium-term enhance the intelligence and reasoning abilities of the system to a greater level. Incor- poration AI agents can query knowledge graphs to complete reasoning tasks, generate hypothesis based on identified graph patterns and relationships, detect knowledge gaps and propose information requirements to a researcher, complete multi-step reasoning through graph paths by answering complex ques- tions, and provide explanations of reasoning chains by using graph evidence in a transparent manner. The query specific subgraphs enable users to get focused subgraphs that relate to a particular research question, save and share subgraph views so as to collaborate, compare subgraphs across document sets and determine the differences, and monitor subgraph evolution across time spans.

The multimodal knowledge graphs include pictures and graphs based on visual entity recognition of papers, audio scripts with speaker extraction of conferences and lectures, video material with scene and object recognition of pre- sentations, tables and structured data integration with the preservation of the information structure of the information, mathematical equations and formulas recognized with the help of OCR and symbolic processing. Long term vision is search- ing high ends that change the way knowledge management is carried out in the world. Unified world knowledge graph seeks to craft interrelated knowledge graph across domains and organizations to enable full coverage, develop federated querying across distributed graphs to allow smooth querying, share ontologies and standardized schemas to allow interoper- ability, develop privacy preserving knowledge sharing systems to protect sensitive data, and develop global knowledge base to be accessible to human and AI systems everywhere to allow democratized access to knowledge.

The continuous learning systems allow self-improving sys- tem architecture, adding published research automatically as it appears, updating graphs by user feedback, corrections to enhance accuracy, learning better extraction patterns to match usage patterns and success metrics, evolving language and terminology with a field, and to have records of past versions to allow reproducibility and knowledge evolution tracking. The cognitive assistance features are offering custom learning paths depending on the knowledge gaps found, computer-generated research hypotheses through finding patterns, predictive views through graph pattern discovery, collaborative building of knowledge to allow distributed contributions and integration with external reasoning systems to give advanced inferences.

X. CONCLUSION

This survey study includes methodologies of knowledge graph construction and retrieval-augmented generation of doc- ument understanding applications. After an intensive examination of nineteen currently existing solutions comprising conversational interfaces, academic search systems, and purposeful graph platforms, we find significant weaknesses in unified fact representation, hallucination mitigation, and explainability of the system. We have found that current systems have 28-39% rates of hallucinating and can only sustain 15% of the necessitated visualization applications. Graph LM fills in on these basic shortcomings with confidence-scored knowledge graphs that allow reliability evaluation, semantic entity extraction pipelines, which are up to 92-95% accurate, and interactive visualization systems, which cater to a wide range of user requirements. The proposed system is based on synergistic technologies to form superior capabilities that are more than the sum of their parts forming realistic avenues in the transformation of raw information into viable knowledge via organized representation and intelligent retrieval systems.

REFERENCES

[1] H. Li, G. Appleby et al., “Knowledge Graphs in Practice: Character- izing their Users, Challenges, and Visualization Opportunities,” IEEE Transactions on Visualization and Computer Graphics, vol. 30, no. 1, pp. 584-594, 2024.

[2] R. Huang et al.,Neo4j Inc., “Neo4j Graph Database Platform Documen- tation,” Neo4j Technical Report, 2024.

[3] D. Edge et al., “From Local to Global: A Graph RAG Approach to Query-Focused Summarization,” arXiv preprint arXiv:2404.16130, 2024

[4] S. Ray, “ChatGPT: A comprehensive review on background, applica- tions, key challenges, bias, ethics, limitations and future scope,” Internet of Things and Cyber-Physical Systems, vol. 3, pp. 121-154, 2023.

[5] J. Zhang et al., “Siren’s Song in the AI Ocean: A Survey on Halluci- nation in Large Language Models,” arXiv preprint arXiv:2309.01219, 2023.

[6] A. Hogan et al., “Knowledge Graphs,” ACM Computing Surveys, vol. 54, no. 4, pp. 1-37, 2021.

[7] J. Johnson, M. Douze, and H. Je´gou, “Billion-scale similarity search with GPUs,” IEEE Transactions on Big Data, vol. 7, no. 3, pp. 535- 547, 2021.

[8] X. Xu et al., “LayoutLMv2: Multi-modal Pre-training for Visually- rich Document Understanding,” in Proc. 59th Annual Meeting of the Association for Computational Linguistics, 2021, pp. 2579-2591.

[9] P. Lewis et al., “Retrieval-Augmented Generation for Knowledge- Intensive NLP Tasks,” in Advances in Neural Information Processing Systems, vol. 33, 2020, pp. 9459-9474.

[10] A. B. Arrieta et al., “Explainable Artificial Intelligence (XAI): Con- cepts, taxonomies, opportunities and challenges toward responsible AI,” Information Fusion, vol. 58, pp. 82-115, 2020.

[11] C. Miao et al.,“Calibrating Knowledge Extraction: A Case Study in Relation Extraction,” in Proc. 2020 Conf. on Empirical Methods in Natural Language Processing, 2020, pp. 3151-3161.

[12] A. Akbik et al., “FLAIR: An Easy-to-Use Framework for State-of- the-Art NLP,” in Proc. 2019 Conf. North American Chapter of the Association for Computational Linguistics, 2019, pp. 54-59.

[13] C. Rasmussen et al., “Schema Evolution in NoSQL Databases: A Systematic Review,” in Proc. 2019 IEEE Int. Conf. Big Data, 2019, pp. 2842-2851.

[14] David Reinsel et al, “The Digitization of the World: From Edge to Core,” IDC White Paper, Doc. US44413318, 2018.

[15] R. Angles et al., “Foundations of Modern Query Languages for Graph Databases,” ACM Computing Surveys, vol. 50, no. 5, pp. 1-40, 2017.

[16] T. H. Nguyen et al., “Relation Extraction: Perspective from Convolu- tional Neural Networks,” in Proc. NAACL Workshop on Vector Space Modeling for NLP, 2015, pp. 39-48.

[17] N. J. Van Eck and L. Waltman, “Software survey: VOSviewer, a computer program for bibliometric mapping,” Scientometrics, vol. 84, no. 2, pp. 523-538, 2010.

[18] E. Curry et al., “The Role of Community-Driven Data Curation for Enterprises,” in Linking Enterprise Data, D. Wood, Ed., Springer, 2010, pp. 25-47.

[19] C. Manning et al., “Introduction to Information Retrieval,” Cambridge University Press, 2008.

[20] B. Lee et al., “Task taxonomy for graph visualization,” in Proc. AVI Workshop on BEyond time and errors: novel evaLuation methods for Information Visualization, 2006, pp. 1-5.