Enhancing Jira Ticket Retrieval with Knowledge Graphs
DISSERTATION

Submitted in partial fulfillment of the requirements of the

Degree: M.Tech in Artificial Intelligence and Machine Learning

By

Boda Dinesh Kumar Reddy

2023AA05149

Under the supervision of

Shivajith Mutteal K

(Engineer, Senior Staff)

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE

Pilani (Rajasthan) INDIA

(May 2025)

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI

SECOND SEMESTER 2024-25

AIMLCZG628T DISSERTATION

Dissertation Title	: Enhancing Jira Ticket Retrieval with Knowledge Graphs
Name of Supervisor	: Shivajith Mutteal K
Name of Student	: Boda Dinesh Kumar Reddy
ID No. of Student	: 2023AA05149

Courses Relevant for the Project & Corresponding Semester:

1. Graph Neural Networks (AIMLCZG514) – Semester 3
2. Conversational AI (AIMLC ZG521) – Semester 3
3. Natural Language Processing (AIMLCZG530) – Semester 2
4. Machine Learning (AIMLCZG565) – Semester 1

Abstract

This dissertation explores the integration of Knowledge Graphs (KGs) into Jira ticket retrieval systems to enhance the performance of existing Retrieval-Augmented Generation (RAG) pipelines. Traditional vector-based RAG systems provide a foundational level of semantic matching but often fail to capture deeper contextual nuances and explicit relationships inherent in Jira tickets. Such limitations lead to imprecise retrievals, where results may be semantically similar in language yet lack the desired contextual relevance.

The proposed approach leverages Knowledge Graph technology to model explicit relationships among ticket elements—such as reporter, assignee, project components, and issue types—thereby creating a richer, unified representation of ticket data. This enhanced structure enables the processing of complex queries (e.g., “tickets reported by X in project Y related to performance issues”) and supports transparent, traceable reasoning that underpins the retrieval process.

The system involves constructing a domain-specific Knowledge Graph from historical Jira data and integrating it with the existing RAG framework. The KG explicitly encodes entity relationships, thereby overcoming the limitations of implicit similarity measures found in pure vector embeddings. The integration is expected to improve precision, enhance explainability, and streamline query processing.

A structured methodology guides the project through eight phases: literature review and landscape analysis, KG schema design, prototype KG construction, system integration with the RAG framework, query interface development, evaluation and feedback collection, system refinement, and final analysis and documentation. The project is scoped for 16 weeks of work, ensuring a rigorous evaluation of performance gains and user impact.

Ultimately, the research aims to deliver a scalable, cost-effective, and on-premise retrieval solution that not only improves ticket matching accuracy but also provides traceable reasoning paths to boost user trust in the system.

Key Words: Knowledge Graphs, Jira Ticket Retrieval, Retrieval-Augmented Generation (RAG), Graph Neural Networks, Natural Language Processing (NLP)

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

II SEMESTER 2024-25

AIMLCZG628T DISSERTATION

Dissertation Outline

BITS ID No. 2023AA05149 **Name of Student:** Boda Dinesh Kumar Reddy

Name of Supervisor	**:** Shivajith Mutteal K
Designation of Supervisor	: Engineer, Senior Staff
Qualification and Experience	**:** MSc Embedded Systems and 15+ years of experience
Official E- mail ID of Supervisor	**:** smutteal@qti.qualcomm.com
Topic of Dissertation	: Enhancing Jira Ticket Retrieval with Knowledge Graphs

B·Dinesh Kumar Reddy

(Signature of Student)

Date: 24 May 2025

(Signature of Supervisor)

Date: 24 May 2025

Project Work Title:

Enhancing Jira Ticket Retrieval with Knowledge Graphs

1.1. Purpose:

The purpose of this dissertation is to enhance the effectiveness of Jira ticket retrieval systems by integrating Knowledge Graphs (KGs) into a Retrieval-Augmented Generation (RAG) framework. Traditional vector-based retrieval methods often fall short in capturing the complex relationships and contextual nuances present in issue tracking data, leading to imprecise and less explainable results. By explicitly modeling entities such as reporters, components, and issue types, and their interconnections through domain-specific KG, the proposed system aims to support more accurate, context-aware, and explainable retrieval. This integration not only improves the handling of complex, multi-entity queries but also fosters greater user trust through transparent reasoning paths, ultimately contributing to faster issue resolution, better knowledge reuse, and a scalable, enterprise-ready solution.

1.2. Expected Outcome:

- Development of a unified Knowledge Graph model tailored to Jira ticket data.
- Integration of the Knowledge Graph with an existing Retrieval-Augmented Generation (RAG) system.
- Enhanced retrieval accuracy through context-aware and relationship-driven search.
- Improved handling of complex, multi-entity queries.
- Increased explainability and transparency in ticket retrieval results.
- Reduction in issue triaging time and improved operational efficiency.
- Prototype system evaluated through user feedback and performance metrics (e.g., MRR).
- Foundation for future extensions such as multilingual support and CRM integration.

2. Literature Review:

Recent advancements in Retrieval-Augmented Generation (RAG) systems have highlighted the limitations of purely vector-based approaches, particularly in domains requiring contextual understanding and explainability. To address these challenges, researchers have explored the integration of Knowledge Graphs (KGs) into RAG pipelines, resulting in improved retrieval accuracy and transparency.

- **Linders & Tomczak (2025)** introduced a KG-extended RAG system that integrates large language models with structured knowledge without requiring additional training. Their approach demonstrated enhanced multi-hop information retrieval and improved answer explainability, particularly on the MetaQA benchmark dataset.
[Source: arXiv: [2504.08893] Knowledge Graph-extended Retrieval Augmented Generation for Question Answering]
- **Sanmartin (2024)** proposed a KG-RAG pipeline that constructs a Knowledge Graph from unstructured text and performs retrieval over the graph. This method significantly

reduced hallucinations in LLM outputs and improved performance on knowledge-intensive tasks.
[Source: arXiv: [2405.12035] KG-RAG: Bridging the Gap Between Knowledge and Creativity]

- **Zhu et al. (2025)** developed the KG^{2}RAG framework, which uses KGs to provide fact-level relationships between document chunks. This approach improved the diversity and coherence of retrieved results, enhancing the overall quality of responses in QA systems.
  [Source: arXiv: [2502.06864] Knowledge Graph-Guided Retrieval Augmented Generation]

- **Xu et al. (2024)** applied a KG-RAG method to customer service question answering by constructing a KG from historical issue tickets. Their system outperformed traditional baselines, achieving a 77.6% improvement in Mean Reciprocal Rank (MRR) and a 28.6% reduction in median issue resolution time.
  [Source: arXiv: [2404.17723] Retrieval-Augmented Generation with Knowledge Graphs for Customer Service Question Answering]

These studies collectively demonstrate the growing consensus in both academia and industry that Knowledge Graphs can significantly enhance the performance, transparency, and usability of retrieval-augmented systems. Their findings provide a strong foundation for applying similar techniques to Jira ticket retrieval, where contextual relationships and traceability are critical.

3.

3.1. Existing Process

The current Jira ticket retrieval system is built on a vector-based Retrieval-Augmented Generation (RAG) framework. This system uses dense vector embeddings to represent both user queries and historical tickets, enabling semantic similarity matching. While this approach provides a foundational level of retrieval performance, it primarily captures surface-level linguistic similarities and lacks the ability to understand deeper semantic structures or relationships between ticket elements. The retrieval process is largely opaque, offering limited insight into why certain tickets are returned as relevant, which can hinder user trust and system refinement.

3.2. Limitations

Despite its utility, the existing system exhibits several critical limitations:

- **Lack of Relationship Encoding**: Vector embeddings do not explicitly represent relationships between entities such as reporter, assignee, components, and issue types, which are crucial for understanding the context of a ticket.

- **Inability to Handle Complex Queries**: The system struggles with multi-criteria queries that involve combinations of entities and conditions (e.g., “tickets reported by user X in project Y related to component Z”).

- **Limited Contextual Understanding**: Without structured knowledge, the system cannot infer or reason about the connections between different ticket attributes, leading to less relevant results.
- **Low Transparency and Explainability**: The retrieval process is difficult to interpret, making it challenging for users to understand why specific tickets were retrieved and how to refine their queries.
- **Reduced Efficiency in Issue Resolution**: The lack of contextual precision often results in the retrieval of irrelevant tickets, increasing the time required for triaging and resolving issues.

4. Justification for Methodology:

The integration of Knowledge Graphs (KGs) into the Jira ticket retrieval process is justified by the need to overcome the inherent limitations of traditional vector-based Retrieval-Augmented Generation (RAG) systems. While vector embeddings are effective at capturing surface-level semantic similarity, they lack the ability to explicitly model relationships between entities, which is critical in the context of issue tracking systems like Jira. KGs provide a structured and semantically rich representation of data, enabling the system to understand and reason over complex inter-entity relationships such as "reported by," "affects," and "belongs to."

This methodology is particularly well-suited for handling complex, multi-entity queries that are common in real-world software development environments. By leveraging KGs, the system can perform precise filtering and retrieval based on structured criteria, significantly improving the relevance and accuracy of results. Moreover, the explicit encoding of relationships enhances the explainability of the retrieval process, allowing users to trace the reasoning behind each result—an essential feature for building trust in AI-driven systems.

The chosen approach also aligns with recent advancements in the field, where KG-augmented RAG systems have demonstrated superior performance in both academic benchmarks and industrial applications. Therefore, the methodology not only addresses the technical shortcomings of existing systems but also positions the solution within a growing body of validated research and best practices.

5. Project Work Methodology:

The project follows a structured, phased methodology distributed across 16 weeks, ensuring focused and incremental development of a Knowledge Graph-enhanced Jira ticket retrieval system. Each phase builds upon the previous one, leading to a fully integrated and evaluated solution.

- **Phase 1: Literature Review & Landscape Analysis**
  Conduct a comprehensive review of academic research and industrial applications of Knowledge Graphs in Retrieval-Augmented Generation (RAG) systems. This phase establishes the theoretical foundation and identifies the best practices for KG construction and integration.
- **Phase 2: Knowledge Graph Schema Design**
  Design a domain-specific schema that captures key Jira ticket entities such as reporter,

assignee, components, and issue types, along with their relationships. This schema serves as the structural blueprint for the Knowledge Graph.

- **Phase 3: Prototype KG Construction**
  Construct a prototype Knowledge Graph using historical Jira ticket data. This involves extracting relevant entities, mapping relationships, and populating the graph using tools like Neo4j or RDF-based frameworks.
- **Phase 4: System Integration with RAG Framework**
  Integrate the constructed Knowledge Graph with the existing vector-based RAG system. This includes modifying the retrieval pipeline to incorporate KG-based reasoning alongside traditional vector similarity.
- **Phase 5: Query Interface Development**
  Develop a user-facing interface that supports complex, multi-entity queries and leverages the Knowledge Graph for enhanced retrieval capabilities.
- **Phase 6: Evaluation & Feedback Collection**
  Evaluate the integrated system using performance metrics such as Mean Reciprocal Rank (MRR), precision, and user feedback. This phase assesses the impact of KG integration on retrieval quality and user experience.
- **Phase 7: System Refinement Based on Feedback**
  Refine the system based on evaluation outcomes. This includes optimizing the Knowledge Graph structure, improving query handling, and enhancing explainability.
- **Phase 8: Final Analysis, Documentation & Presentation Preparation**
  Summarize the project findings, document the methodology and results, and prepare the final dissertation report and presentation materials.

6. Benefits Derivable from the Work:

The integration of Knowledge Graphs into Jira ticket retrieval systems offers several tangible and strategic benefits:

- **Improved Ticket Matching Accuracy**: By explicitly modeling relationships between ticket entities, the system retrieves results that are more contextually relevant and semantically precise.

- **Faster Issue Triaging and Resolution**: Enhanced retrieval precision reduces the time spent by users filtering through irrelevant tickets, thereby accelerating the issue resolution process.

- **Explainable and Transparent Retrieval**: The use of Knowledge Graphs enables traceable reasoning paths, allowing users to understand why specific tickets were retrieved, which builds trust in the system.

- **Enhanced Knowledge Reuse**: Unified graph-based representations of tickets facilitate the discovery of related issues and solutions, promoting better reuse of organizational knowledge.

- **Support for Complex Queries**: The system can handle multi-entity and relationship-based queries that are difficult to process using traditional vector-based methods.

- **Scalability and Extensibility**: The modular design allows for future integration with CRM systems, multilingual support, and domain-specific extensions, making the solution adaptable to evolving enterprise needs.

7. Additional Details:

To ensure long-term relevance and adaptability, the proposed system is designed with extensibility as a core principle. The architecture supports domain-specific customization, allowing the Knowledge Graph to be tailored to various departments or business units within the organization, where ticket metadata and terminology may differ significantly.

Furthermore, the system is engineered with on-premises deployment capabilities to address data privacy, security, and compliance requirements. This is particularly important for enterprises operating under strict data governance policies. These extensibility and deployment features collectively position the solution as a scalable, secure, and future-ready enhancement to traditional Jira ticket retrieval systems.

1. Broad Area of Work:

This project falls under the broad areas of **Information Retrieval**, **Knowledge Representation**, and **Semantic Search**. It intersects with the fields of Natural Language Processing (NLP), Explainable Artificial Intelligence (XAI), and Graph-Based Machine Learning, focusing on enhancing the interpretability and contextual relevance of search systems in enterprise environments.

2. Objectives:

- To design and construct a domain-specific Knowledge Graph that models key entities and relationships in Jira ticket data.
- To integrate the Knowledge Graph with an existing Retrieval-Augmented Generation (RAG) system for enhanced ticket retrieval.
- To enable complex, multi-entity query processing with improved contextual understanding.
- To improve the explainability and transparency of the retrieval process through traceable reasoning paths.
- To evaluate the system's performance using metrics such as Mean Reciprocal Rank (MRR), precision, and user feedback.
- To demonstrate the feasibility of deploying the solution in a scalable, on-premise enterprise environment.

3. Scope of Work

The scope of this project encompasses the design, development, and evaluation of a Knowledge Graph-enhanced Jira ticket retrieval system. It includes the following key components:

- **Knowledge Graph Design and Construction**: Define a schema and build a domain-specific Knowledge Graph that captures entities such as reporters, assignees, components, and issue types, along with their relationships.
- **Integration with RAG Framework**: Modify the existing vector-based Retrieval-Augmented Generation system to incorporate KG-based reasoning and filtering, enabling hybrid retrieval.
- **Complex Query Handling**: Enable the system to process and respond to multi-entity, relationship-driven queries with improved contextual relevance.
- **Explainability and Transparency**: Implement mechanisms to trace and explain the reasoning behind retrieved results using the graph structure.
- **Evaluation and Feedback**: Conduct performance evaluation using metrics like Mean Reciprocal Rank (MRR), precision, and user feedback to assess improvements in retrieval quality and user satisfaction.

- **Scalability and Extensibility**: Ensure the system is designed for future enhancements, including multilingual support, CRM integration, and domain-specific adaptations.

4. Detailed Plan of Work

S. No.	Tasks / Subtasks	Start Date – End Date	Planned Duration (weeks)	Specific Deliverable
1	Literature Review & Landscape Analysis	Week 1 – Week 2	2	Literature review report
2	Knowledge Graph Schema Design	Week 3 – Week 4	2	KG schema document
3	Prototype KG Construction • Entity extraction • Relationship mapping • Graph population	Week 5 – Week 7	3	Populated prototype Knowledge Graph
4	System Integration with RAG Framework	Week 8 – Week 9	2	Integrated KG-RAG system
5	Query Interface Development	Week 10 – Week 11	2	Functional query interface
6	Evaluation & Feedback Collection	Week 12 – Week 13	2	Evaluation report with performance metrics
7	System Refinement Based on Feedback	Week 14 – Week 15	2	Refined system with improved retrieval and explainability
8	Final Analysis, Documentation & Presentation Preparation	Week 16	1	Final dissertation report and presentation slides

5. Literature References:

1. Linders, M., & Tomczak, J. (2025). *Knowledge Graph-Extended Retrieval-Augmented Generation for Multi-Hop Question Answering*. arXiv preprint. https://arxiv.org/abs/2504.08893

2. Sanmartin, A. (2024). *Reducing Hallucinations in LLMs with Knowledge Graph-Augmented Retrieval Pipelines*. arXiv preprint. https://arxiv.org/abs/2405.12035

3. Zhu, Y., et al. (2025). *KG^{2}RAG: Knowledge Graph-Guided Retrieval-Augmented Generation Framework*. arXiv preprint. https://arxiv.org/abs/2502.06864

4. Xu, L., et al. (2024). *Enhancing Customer Service QA with Knowledge Graph-Augmented RAG Systems*. arXiv preprint. https://arxiv.org/abs/2404.17723

6. Supervisor’s Rating of the Technical Quality of this Dissertation Outline

EXCELLENT / GOOD / FAIR/ POOR (Please specify): GOOD

Name of the supervisor: Shivajith Mutteal

Email Id of Supervisor: smutteal@qti.qualcomm.com

Mob # of supervisor: 8884844448

(Signature of Supervisor)

Date: 24 May 2025