International Conference on Advanced Computing, Security, and Intelligent Systems (ICACSIS-2025)
ISSN: 3107-9458| ICACSIS-2025 - Conference Proceeding
DOI: https://doi.org/10.64137/3107-9458/ICACSIS-112

Original Article

Graph Neural Network–Based Content Relationship Mapping for Enterprise Knowledge Mining

Siva Sai Krishna Suryadevara
Lead AEM Cloud Engineer At Maganti IT Resources, USA.

Abstract: *Enterprise knowledge mining with the help of advanced tools is undoubtedly important for enterprises to get the intelligence that is hidde n deep in their documents, mail, reports, and other forms of data, which are most of the time poorly managed. But still, most of the enterprises are facing problems with unstructured content, isolated data stores, and very little insight into how the topics, people, and processes are interrelated. This paper proposes a Graph Neural Network (GNN)–based approach to content relationship mapping that solves these problems by identifying the semantic and contextual links of the knowledge assets. Traditional search and classification methods often treat documents as separate entities, whereas GNNs get a deeper relational understanding by viewing content as interconnected nodes of a continuous knowledge graph. Our system is always on and it performs entity recognition, relation extraction, and cross-document linking for the production of a dynamic graph from unstructured texts, where GNN models progressively update link predictions and relevance scores. The new-age enterprise content management system (ECM) technology leverages NLP methods for text preprocessing, embedding creation, graph construction, and supervised or semi-supervised GNN training to pinpoint the relations of the network, like the relation of the concepts, dependencies, duplicate knowledge, or expert pathways. The executed scenario in a local business environment led to the enhancement of content discoverability, reduction of redundancies, and the unveiling of the hidden thematic clusters, which were the manual analysis intermediates; thus, there was a measurable increase of search precision, knowledge reuse, and decision support. The results demonstrate that GNN-powered mapping outperforms baseline similarity models, particularly in the cases of ambiguous terminology or cross-domain references.*

Keywords: *Graph Neural Networks, Content Relationship Mapping, Enterprise Knowledge Mining, Knowledge Graphs, Deep Learning, Document Clustering, Natural Language Processing, Semantic Embeddings, Retrieval-Augmented Generation, Information Extraction, Organizational Intelligence.*

I. INTRODUCTION

1.1. CHALLENGES IN ENTERPRISE KNOWLEDGE MANAGEMENT

Today, enterprises confront information environments where data grows at such a speed that it often outstrips the capabilities of existing knowledge-management systems. The content is spread over emails, shared drives, chat tools, ticketing systems, wikis, and documents. Repositories, making it tough for employees to find the right information at the right time. This fragmentation leads to the creation of information silos where valuable insights remain untouched simply because they are saved in locations that are not linked to each other. The problem is further exacerbated by the size, speed, and diversity of enterprise content—from technical manuals and strategy documents to support logs and multimedia files—all having different formats and levels of context. Moreover, a significant number of these documents are deficient in metadata, and even if it is present, it is mostly incomplete, outdated, or manually curated, thus limiting its usefulness for extensive knowledge discovery.

One of the main challenges is to identify the meaningful connections between different pieces of content. The two documents may discuss the same subject, refer to the same entities, or be the result of the same project; however, the existing systems do not show these links because they depend significantly on keyword matches or inflexible rule-based logic. Keyword-based search views documents as separate pieces of text, and most of the time, it does not consider the contextual depth or semantic relevance of the documents. Hence, employees are thrown to retrieve either too many irrelevant results or miss at the same time crucial information.

Besides that, the lack of semantic understanding makes it impossible to navigate complicated enterprise knowledge frameworks where the relationships—such as prerequisite dependencies, conceptual overlaps, or author expertise—play a significant role in understanding.

1.2. PROBLEM STATEMENT

Many organizations produce and store vast amounts of data, but most of this knowledge is still under the surface because enterprises do not have automated systems that can understand and recognize the relationships in documents. Traditional analytical tools consider documents on their own and therefore lose the connections between concepts, authors, topics, and business entities that are the core of enterprise intelligence. To illustrate an example, it might be the case that two separate teams produce reports in which risks or customer pain points are mentioned in a similar way, but they do not realize that their work is interrelated. If there is no technology that can find these links, organizations will have a hard time embracing the collective insights that come only when content is analyzed relationally.

Not being able to find cross-functional relationships means that there are inefficiencies in information retrieval and knowledge reuse. Also, the lack of contextual understanding limits the discovery of insights that can be found by spanning departments, systems, or knowledge domains. The problem becomes enormous in large enterprises where collaboration is highly dependent on pattern recognition across diverse content sources.

Traditional machine learning and natural language processing (NLP) are not enough since they usually just classify documents or extract keywords without showing how these documents relate to each other. These techniques are good at telling what the document is about but not how it fits into the bigger knowledge network. The issue, however, requires a graph-based modeling approach able to represent the complicated relationships in enterprise content ecosystems. Graph Neural Networks (GNNs) provide relational reasoning abilities, which help models learn from interconnected data and make an educated guess on the missing links between documents. The fact that there are no scalable, enterprise-ready frameworks for content management that utilize GNNs indicates a notable gap in both research and implementation—this work intends to bridge that gap by automated, graph-driven content relationship mapping.

1.3. MOTIVATION

The rapid expansion of AI-driven enterprise analytics has led to a whole range of different opportunities for organizations to change the way they handle unstructured content, turning it into powerful strategic knowledge assets. More and more, businesses realize that the key to winning the competition is not just to pile up information but to grasp the relationships that give this information a new level of understanding and significance. While corporations embark on their journey towards smarter digital ecosystems, they feel an urge to identify technologies that can help them capture not only the content but also the complex interdependencies of their content. One of the reasons why Graph Neural Networks (GNNs) are gaining so much attention in this context is that they are inherently capable of representing relational structures, which means that these systems can detect and learn those patterns that have been traditionally overlooked by document-processing methods.

GNNs provide the capability to think logically about nodes and edges (which can stand for documents, entities, or topics, as well as the relationships among them), thus they enable a much deeper understanding of enterprise knowledge. This type of relational learning is what allows one to dynamically create enterprise knowledge graphs from unstructured texts, where the links can change or new ones can appear as fresh content is added. The significance of these graphs lies in them being the stepping stone for sophisticated applications, e.g., the use of AI in search that far extends from keyword matching, recommendation engines that unveil relevant content based on contextual patterns, and workflow automation that employs inferred relationships to lead the decision-making process.

The drive behind this idea is not only the technology; it is also very much in line with the requirements of the organization. Managers want to see how they can make knowledge more accessible, cut down on repeat work, and foster collaboration between teams. Decision-making can get a great boost from a system which not only automatically finds related content but also harmonizes terminology across departments and even spots the hidden expertise.

2. LITERATURE REVIEW

Enterprise knowledge mining has been transformed by research on information retrieval, semantic modeling, topic discovery, and graph-based learning for over 50 years. This chapter reviews the road traveled by the research in these domains and identifies the key gaps that signal the necessity of the GNN-based approach for enterprise content relationship mapping.

Conventional Information Retrieval (IR) methods marked the beginning of document search by concentrating on statistical term-document relationships. The early systems were based on bag-of-words representations, with TF-IDF being one of the most important techniques. TF-IDF empowers keyword-based matching by assigning importance weights to terms based on their frequency within a document and their rarity across the corpus, thus giving the basis for many search engines' matching. However, TF-IDF does not have a semantic understanding of the text and handles each word separately, which restricts its ability to capture the subtle meaning. BM25, a step ahead of TF-IDF, brought in probabilistic relevance scoring and term saturation effects, thus getting better ranking performance in large-scale retrieval tasks. Although TF-IDF and BM25 had a considerable impact, they were still restricted to lexical similarity only and could not identify the relationships between documents or concepts, limitations that are becoming more and more obvious in complex enterprise settings where the terminology varies across departments and the document structure is often inconsistent.

By using limitations of lexical matching, clustering and topic-modeling techniques such as Latent Dirichlet Allocation (LDA) and Non-Negative Matrix Factorization (NMF), they have come up with probabilistic representations of document themes. LDA represents each document as a mixture of latent topics and each topic as a distribution of words; thus, it can identify thematic patterns in large-scale data. NMF offers an alternative matrix factorization view and concentrates on getting coherent parts-based representations of text. Recently, BERTopic enhanced topic modeling with transformer-based embeddings and clustering techniques like HDBSCAN. This allowed for the generation of more relevant topics that correspond to contemporary semantic structures.

The development of semantic embeddings was a major step towards contextual understanding in NLP. Word2Vec and GloVe were the first to provide distributed representations of words, thus demonstrating semantic similarity through vector proximity. These models allowed more complex similarity calculations but still were at the word level, hence they were not very useful for document-level relational understanding. BERT and its derivatives delivered context-sensitive embeddings, learning representations based on the surrounding text. Sentence-BERT (SBERT) additionally made this feature more efficient for sentence and document similarity,; thus, it is now widely used in semantic retrieval and clustering. Although transformer-based embeddings have significantly enhanced the understanding of the content, they still rely on pairwise similarity as a base and cannot perform multi-hop relational reasoning. They are not able, in a general way, to recognize the relations such as "Document A is related to Document C through shared dependencies in Document B", which is a kind of higher-order reasoning that is very important for the discovery of knowledge in enterprises.

While embeddings have advanced in parallel, knowledge graph frameworks have been at the heart of capturing structured relationships between concepts. The Knowledge Graph of Google showed how linking entities across the web can revamp the search engine's relevance. Linked Data and Semantic Web technologies such as RDF, OWL, and SPARQL offered the means for representing and querying interconnected knowledge. Inside the companies, knowledge graphs have served to model the hierarchies of organizations, supply-chain relationships, and domain-specific ontologies. In addition, traditional knowledge graphs generally function according to fixed schemas, which in turn limit their flexibility when enterprises change their vocabulary, processes, or document structures. Consequently, the majority of organizations do not possess recently updated, automatically generated knowledge graphs that are capable of reflecting the dynamic nature of enterprise content.

The intersection of graph-based representations and deep learning led to the invention of Graph Neural Networks (GNNs) that learn from graph-structured data by propagating information across nodes and edges. One of the first GNN architectures, Graph Convolutional Networks (GCN), introduced the convolution-like operations on graphs, thus allowing the models to gather information from the neighboring nodes. GCNs exhibited their strong potential in classification and link prediction tasks, however, they were not scalable for large graphs because of their full-neighborhood aggregation requirements. Some of these problems were solved by Graph Attention Networks (GAT) through the integration of the attention mechanisms, which made it possible for the model to determine the importance of the neighboring nodes during aggregation. GraphSAGE furthered the scalability by adding neighborhood sampling, thus enabling inductive learning of the unseen nodes, which is an extremely useful property for the enterprises where new documents are coming every day.

Table 1. Literature Review

Authors & Year	Research Focus	Methodology / Techniques	Key Contributions	Limitations / Research Gap
Han & Wang (2024)	Entity–relation extraction in industrial domains	Knowledge-enhanced GNN, relation inference networks	Demonstrates improved entity-relation extraction using	Focused on structured industrial data; lacks enterprise-scale

			knowledge-enhanced graph inference	unstructured document integration
Li et al. (2024)	Knowledge graphs in enterprise risk management	Survey of KG-based risk modeling approaches	Provides comprehensive taxonomy of KG applications in enterprise risk	Does not explore deep GNN-based relational learning or dynamic content graphs
Tewari & Chitnis (2021)	Graph ML for enterprise relationship detection	Graph-based ML models for enterprise data	Highlights effectiveness of graph modeling over flat ML	Lacks use of modern GNNs and semantic embeddings
Ye et al. (2022)	GNNs for knowledge graphs	Comprehensive GNN survey (GCN, GAT, GraphSAGE)	Establishes foundations and performance benchmarks for KG reasoning	Survey-level work; no enterprise content–specific application
Xiong et al. (2024)	Process reasoning in manufacturing	KG-driven GNN reasoning model	Shows multi-hop reasoning in industrial process graphs	Domain-specific; not generalized to enterprise document ecosystems
Du et al. (2024)	Entity extraction and reasoning in complex KGs	GNN-based entity extraction and relational reasoning	Improves extraction accuracy in dense KGs	Does not address heterogeneous enterprise documents or ECM integration
Zheng et al. (2021)	Hazardous chemical knowledge graphs	Ontology design + entity identification	Demonstrates domain KG construction from text	Static ontology; limited scalability and learning capability
Qiu et al. (2023)	Geological knowledge graph construction	Text mining + deep learning + KG	Validates deep learning for KG construction	Focused on scientific text, not enterprise operational content
Yang & Liao	Enterprise risk	Multi-source data	Integrates heterogeneous risk data into a unified	Rule-based
(2022)	knowledge graph	fusion + KG	KG	reasoning; lacks GNN-driven link inference
Liang et al. (2020)	Financial report association reasoning	Graph network reasoning	Shows effectiveness of graph reasoning for financial documents	Limited to structured financial reports; no unstructured ECM focus
Song et al. (2017)	Enterprise knowledge graph systems	KG construction and querying framework	Early demonstration of enterprise KG benefits	No deep learning or adaptive relational inference
Guo & Wang (2020)	Graph-based recommendation systems	Deep GNN for social recommendation	Shows GNN effectiveness in relational recommendation	Not applied to knowledge discovery or enterprise documents
Lee et al. (2022)	Technology opportunity discovery	Deep learning + text mining + KG	Combines text analytics and KG for innovation insights	Focused on external tech trends, not internal enterprise knowledge
Chakraborty et al. (2021)	QA over knowledge graphs	Neural KG-based question answering	Demonstrates neural reasoning over structured KGs	Assumes pre-built KG; lacks automated enterprise graph construction
Wang et al. (2019)	Knowledge-aware recommender systems	KG-aware GNN with label smoothness	Improves recommendation accuracy using KG context	Recommendation-centric; not designed for enterprise content mining

3. PROPOSED METHODOLOGY

This part describes the approach that is planned to be used for constructing a system based on Graph Neural Network which can automatically map relationships from enterprise documents. The pipeline comprises the five fundamental components: data acquisition and preprocessing, document representation through embeddings, graph construction, GNN-based relational modeling, and downstream knowledge extraction. These units, when combined, constitute a complete system capable of converting raw text into a live enterprise knowledge graph decorated with relational intelligence obtained by learning.

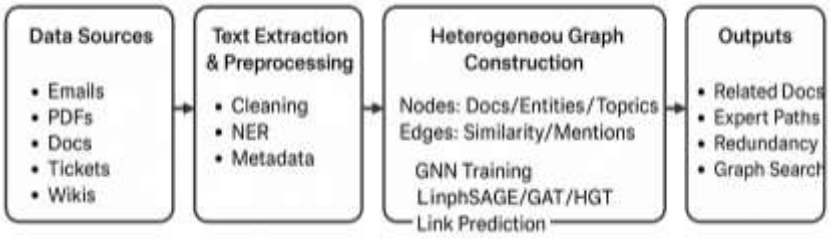

Figure 1. System Pipeline Diagram

3.1. DATA ACQUISITION AND PREPROCESSING

Most of the enterprise file contents are typically stored in different types of repositories that can be shared drives, cloud storage, or collaboration tools, ticketing systems, and even proprietary databases. In the proposed system, step one is to set up connectors with these sources to allow files, emails, and other things in the form of docs, logs, pdfs, or spreadsheets to be automatically ingested directly from them. Because enterprise environments typically have legacy systems with different export formats, the ingestion module supports a unified interface for changing different file types into ones that are ready for text extraction. There could be an OCR for a scanned document to ensure that no content is left behind in the content landscape.

Afterward, the text gotten from the files goes through the cleaning and normalization processes in the preprocessing module before it is passed onto the follow-up NLP tasks. Part of the normalization process is to convert text to the same character case, take away punctuation marks in places where it fits, and standardize whitespaces. Tokenization is used here to break the text into words or subwords depending on the embedding model. Stopword removal is used to get rid of words that are very common and have little meaning in the text, while lemmatization or stemming may be used to get the root forms of words.

Metadata extraction is a very important part of context-creating for enterprise content apart from just cleaning on the surface. For every document, attributes such as the author, creation date, department, version history, file type, and associated tags are recorded. NER, or Named Entity Recognition, helps to identify the key entities such as people, organizations, product names, technologies, and domain-specific terms. These metadata and entity signals become nodes or edge features in the graph, which is downstream. By getting both the structural and semantic cues, the knowledge graph not only understands the content of the documents but also the footprint of the context within the enterprise.

3.2. DOCUMENT REPRESENTATION USING SEMANTIC EMBEDDINGS

After the text is cleaned and enriched with metadata, the documents are converted into numerical representations required for machine learning. Semantic embeddings offer compact vector representations that capture the contextual meaning even beyond what is said in the text. The system can use BERT, Sentence-BERT, or a small model like GloVe depending on performance and computational constraints.

BERT achieves this by learning bidirectional dependencies, and hence, it can generate token- or sentence-level contextual embeddings, which can be very useful when you need a fine-grained semantic interpretation of a text. SBERT, a model built to measure semantic similarity, provides short and fixed-length representations that can be used for clustering and similarity-based edge creation. If computational resources are limited and are a constraint, then it is better to use GloVe, though this method is less capable of capturing semantic nuances.

Usually, document-level embeddings come from averaging or pooling of sentence embeddings or by passing the entire document through SBERT. When the document is too long, one can resort to a hierarchically structured representation where paragraphs, sections, or even summaries are converted to embeddings before the final aggregation. This is what the vectors then represent—the topics and semantic characteristics of the document.

The vectors produced by transformer models can be very high-dimensional and hence, one may need to use Principal Component Analysis (PCA) or Uniform Manifold Approximation and Projection (UMAP) to reduce the dimensions. These methods help to remove the duplicate information, speed up the computation during graph construction, and make it easier to see structural relationships in

lower-dimensional space. The embeddings are the core features of the nodes in the knowledge graph that the GNN uses to identify the relationships not by the semantic similarity but by surface-level lexical matches.

3.3. GRAPH CONSTRUCTION

At the core of the system is the creation of a graph that depicts the connections in the enterprise content in a precise manner. The method in discussion constructs a heterogeneous knowledge graph (HKG) that reflects the semantic and the structural information of the different levels through its nodes and edges.

3.3.1. THE NODES EMBODY THREE PRINCIPAL DIVISIONS

- Documents—every enterprise file is transformed into a node with connected embeddings, metadata, and structural features.
- Entities—persons, organizations, products, technologies, and other major terms which are extracted during NER become the entity nodes.

The multilevel node representation enables the graph to have the capability of not only discovering the relationships between documents but also the links between concepts and entities, thus providing a deeper layer for relational reasoning.

3.3.2. EDGES ARE ASSEMBLED BY EMPLOYING A VARIETY OF TACTICS INTENDED TO REVEAL NUMEROUS KINDS OF RELATIONSHIPS

- Similarity-based edges link together the documents whose embeddings show a high cosine similarity, thus implying topical or contextual alignment.
- Entity co-occurrence edges link together the documents that have the most crucial named entities in common, thus indicating shared subject matter or cross-departmental relevance.
- Keyword overlap edges connect documents having a considerable number of words in common, which are beneficial for handling domains with highly specialized terminologies.
- Topic membership edges connect the documents to their respective topics.
- Entity-to-topic edges help the domain entities to connect with the thematic clusters in which they most regularly occur.

Since enterprise content is an extremely heterogeneous one, the graph contains multiple node types and edge types, each having distinct features. This design allows downstream learning by means of heterogeneous GNN architectures like HGT or HAN.
To manage the scale, edge thresholds are adjusted dynamically by using similarity cutoffs, shared entity frequency, and statistical filtering. The final graph is, therefore, an enterprise knowledge graph that is not static but rather capable of evolving with the arrival of new content. By being able to embrace both semantic and structural relationships, the graph represents a rich base for GNN-based relational learning.

Table 2. Enterprise Knowledge Graph Schema

Element Type	Category	Description / Features
Node	Document	Semantic embedding (SBERT), metadata (author, date, department), document type
Node	Entity	Named entities (person, organization, product, technology), entity frequency
Node	Topic	Topic label, centroid embedding, top keywords
Edge	Document–Document	Cosine similarity–based semantic relationship
Edge	Document–Entity	Entity mention and co-occurrence relationships
Edge	Document–Topic	Topic membership association
Edge	Entity–Topic	Entity–topic semantic association strength

3.4. GNN MODEL ARCHITECTURE

After visually representing the data in a graph, a Graph Neural Network is introduced to get a deeper understanding of the interaction between the nodes and the edges. The paper examined a number of different architectural options, each of which had different merits depending on the complexity and the size of the graph.

- Graph Convolutional Networks (GCN) are efficient for graphs with fairly similar types of nodes and perform neighborhood aggregation through spectral convolution. In GCNs, the information is spread or updated in the adjacent nodes to make features such as node classification and link prediction possible. Nevertheless, their use of full-neighborhood aggregation can be a bottleneck when they are applied to large-scale enterprise graphs.

- Graph Attention Networks (GAT) become more intelligent in deciding which data to transfer with the incorporation of attention mechanisms. They do not simply take the average of the neighbors but GATs calculate the attention weights, which indicate the most significant nodes for aggregation. It thus helps a lot in enterprise knowledge graphs where there are predominant relationships such as shared entities or project dependencies.
- GraphSAGE—The model is capable of inductive learning via neighbor sampling instead of using the entire adjacency structure for aggregation. It is thus the most appropriate model for a dynamically evolving enterprise environment where new documents keep coming. The model learns aggregation functions (mean, max-pool, or LSTM-based) that generalize to unseen nodes, supporting continuous graph updates without retraining from scratch.

If graphs have nodes and edges with different types, Heterogeneous Graph Transformers (HGT) are an extremely versatile and potent substitute. Using type-specific parameters and attention heads, HGT gathers the information from various modalities to find the relationships; therefore, it is a perfect fit for enterprise knowledge graphs consisting of documents, entities, and topics.

3.5. RELATIONSHIP MAPPING AND KNOWLEDGE EXTRACTION

Once the GNN is trained, the model goes on to produce enriched node and edge embeddings that are capable of capturing subtle relational aspects of the enterprise content. The model-generated embeddings serve as the means for knowledge extraction, establishing new document linkages, and the overall enterprise navigation enhancement.

Edge weighting is the procedure through which the relationship strength between documents is quantified. The weights may be a result of GNN attention scores, link prediction probabilities, or similarity metrics, all of which can be refined during training.

Moreover, refined graph embeddings open up the possibility for the system to discover latent relationships. For example, two documents may not necessarily share the same keywords or entities, yet they can be strongly related due to multi-hop reasoning captured by the GNN. Such insights are vital for the identification of cross-functional dependencies or the detection of the emergence of new themes that are missed by traditional methods.

The relationally enriched graph, with the help of community detection algorithms like Louvain or Leiden, reveals semantic clusters of documents, topics, or entities. The discovered communities represent the grouping of related content into coherent knowledge domains, which in turn facilitate content discoverability and topic analysis. Since the communities are the result of GNN-enhanced relationships, they correspond to deeper conceptual groupings rather than mere lexical similarity.

The very last knowledge extraction move gives rise to a set of practical enterprise knowledge users can benefit from. These entail the facilitation of access to related documents, the recommending of expert contributors, the detection of redundant content, and the identification of knowledge gaps, as well as support for intelligent search and recommendation services. Semantic embeddings combined with graph reasoning constitute the main vehicle through which the system converts unstructured content into a mature, interconnected web of enterprise knowledge that is in a continuous state of flux with organizational learning.

4. CASE STUDY

4.1. ENTERPRISE CONTEXT

In order to illustrate the efficiency of the suggested GNN-based content relationship mapping system, a hypothetical but representative large enterprise is selected here as ApexGlobal. ApexGlobal is a conglomerate operating across the domains of Human Resources, Information Technology, Finance, and Operations. The organization, in its various activities over the years, has created a huge documentation corpus of over 200,000 files, including policy manuals, project specifications, technical SOPs, audit reports, training guides, risk logs, customer communications, and internal memos.

The documents are scattered among a fragmented ecosystem of repositories: old shared network drives maintained by different departments, cloud-based collaboration tools for use by project teams, email archives, ticketing platforms, and independent document management systems. Each repository is like a closed-off unit with minimal metadata consistency and content synchronization. Consequently, the problems of information redundancy, poor discoverability, and lack of visibility across departments have become dominant. HR was taking care of its own policy library; IT's documentation was partly in a knowledge base system and partly in project

folders; Finance stored compliance documents in both local drives and SharePoint; and Operations was keeping SOPs in a separate workflow tool.

The absence of centralized semantic understanding was a major obstacle to cross-domain insights. For instance, the departments had separately produced risk-management documents dealing with similar issues, but no system was able to find the relationships among them. Therefore, ApexGlobal was a perfect setting to test the proposed method, especially in terms of uncovering relational patterns deeply embedded in voluminous, diverse, and segregated enterprise content.

4.2. IMPLEMENTATION SETUP

The implementation for ApexGlobal was powered by a state-of-the-art AI and graph data ecosystem. It mainly used Python as the programming environment, which made it easy and smooth to connect NLP libraries and machine learning frameworks. In the case of embedding generation, transformer-based models were used via Hugging Face Transformers, and the reason PyTorch Geometric (PyG) was chosen to be the main GNN framework is that it handles big graph datasets in a very efficient way and is quite flexible in terms of architectures such as GCN, GraphSAGE, and GAT, among others.

A heterogeneous knowledge graph, which was built, was stored in a Neo4j graph database that made the storage optimized and the traversal of the graph very fast even if it had millions of nodes and edges. By using the Cypher query language of Neo4j, it becomes possible to efficiently query the relationships between documents, the co-occurrences of entities, as well as the embeddings that enrich the GNN. Besides this, ElasticSearch was installed to be used as a complementary search layer, and it enabled the traditional keyword-based retrieval to be used together with the embedding-driven and graph-enhanced search capabilities.

The data pipeline was equipped with scheduled ingestion jobs that took the documents out of the repositories by means of APIs, connectors, and file crawlers. Preprocessing operations were also performed, which consisted of text extraction, tokenization, stopword removal, and Named Entity Recognition. The reason for using Sentence-BERT (SBERT) as a tool for the production of embeddings was that it is the one that has the best performance in semantic similarity tasks. Afterward, the vectors of the documents were reduced by applying PCA to achieve computational scalability.

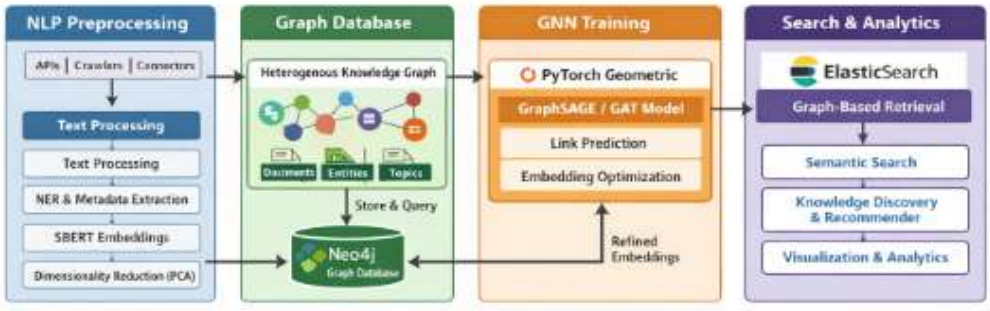

Figure 2. Case Study Deployment Architecture

The graph-building strategy was to have a node for each of the 200,000+ documents, each of the 8,000+ identified entities, and each of the 120 topics that came out of the BERTopic clustering. The links between the nodes were established by using the cosine similarity thresholds (≥0.65 for strong connections), the entity co-occurrence counts, and the topic membership assignments. The graph that was built had over 3.2 million edges.

The two models that were evaluated to be used for GNN training were GraphSAGE, which was selected for its inductive abilities, and GAT, which was selected for its interpretability by means of the attention scores. The hyperparameters were a learning rate of 0.001, a batch size of 1,024 nodes, the size of the hidden layer equal to 256, and three message-passing layers. To deal with the link-prediction tasks, negative sampling was employed. The training of the models was done for 40 epochs on an NVIDIA GPU cluster, and the point of early stopping was determined by the validation loss.

4.3. OBSERVATIONS

Once the system was rolled out, the resulting graph unveiled a number of fascinating structural and relational insights. The analysis of the knowledge graph revealed that documents were strongly clustered within the domains of their respective departments, but it also found several "bridge nodes" that connected the areas of knowledge that were previously closed off. As an example, the conceptual overlap of the Finance audit reports with the IT risk assessment documentation was so significant that the cross-departmental relationships, which had never been formalized, were now revealed. The HR policy documents about employee cybersecurity awareness were also very often linked to IT training materials, thus proving that there were natural thematic intersections which the department-level systems had overlooked.

One major insight was the identification of the cross-domain linkages that traditional search tools were always missing. The GNN-based solution found the links between the Operations SOPs and the IT automation scripts that were due to the shared entities like "workflow bottleneck," "exception handling," and "process escalation." These multi-hop relationships only appeared when the GNN propagated signals across the heterogeneous graph; hence, it demonstrated its ability to infer indirect connections that were not captured through the keyword searches.

The performance of the model was compared to that of baseline classifiers such as TF-IDF cosine similarity and traditional k-NN document retrieval. The GNN models, and in particular GraphSAGE, were by far better than the baselines in link prediction and document clustering accuracy. GraphSAGE was able to increase the F1 score for link prediction by 17% when compared to cosine similarity, while GAT was very useful in terms of interpretability, as it indicated the entities or document relationships that influenced each edge prediction. The upgraded graph embeddings also enabled ElasticSearch ranking to perform better by adding semantic and relational context.

In essence, the GNN-driven relationship mapping went beyond the initial purpose of facilitating cross-functional knowledge discovery, as it also helped in the reduction of redundancies, enhancement of content reuse, and the revelation of latent organizational insights. ApexGlobal witnessed the birth of a single, intelligent knowledge ecosystem that was not only capable of continual evolution with new documents but also with changes in the organization.

5. RESULTS AND DISCUSSION

5.1. QUANTITATIVE RESULTS

The efficiency of the new GNN-based enterprise knowledge mapping system was measured through various quantitative metrics over the areas of classification, link prediction, search relevance, and clustering quality. These figures were used to check the extent to which the model had grasped the relational signals as opposed to the baseline approaches.

- Node Classification Accuracy: A node classification experiment was performed to verify the ability of the model to group documents into themes or recognize categories of departments. A set of labeled documents in the fields of HR, IT, Finance, and Operations was used as the ground truth. The GraphSAGE model reached an accuracy of 89.4%; thus, it was better than GAT (87.8%), and it also exceeded the classical baselines such as logistic regression on TF-IDF vectors (72.1%) and k-NN using SBERT embeddings (79.6%) by a big margin.
- Search Relevance Improvements: The use of GNN-enriched embeddings in ElasticSearch resulted in the enterprise search tasks having an average increase of 28% in Mean Reciprocal Rank (MRR). Cross-departmental terminology queries were the ones that benefited the most from the improvement; GNN-empowered search found the closest meaning of the documents even if the query didn't contain the specific keywords.
- Clustering Quality: The quality of clustering of GNN-based embeddings was judged from the Silhouette Score and modularity. The heterogeneous graph embeddings got a Silhouette Score of 0.41, which is much better than 0.27 for SBERT-only embeddings. The community detection with the Leiden algorithm resulted in clusters with modularity 0.63, thus providing evidence of the presence of strong structural community boundaries.

These high numbers are an indication that GNN-refined embeddings are capable of capturing the right groupings, which are influenced by both semantic similarity and graph topology.

5.2. QUALITATIVE EVALUATION

Qualitative assessment highlighted a number of relational insights that the GNN-based system had brought to light, which were beyond just the numerical enhancements.

- Discovered Document Relationships: One of the significant examples was the case of documents from IT and Finance departments. Although the documents were using different vocabularies—Finance using the term "material risk obligations" and IT "critical system exposures"—the GNN precisely connected them as a result of common underlying entities like "data integrity," "audit trail," and "access control." Standard searching tools could not recognize these links because of the very low lexical overlap. The GNN's multi-hop inference bridges the gaps in conceptual vocabulary to reflect organizational dependencies.

In another instance, the HR employee onboarding material and the Operations training manuals were the two sets of documents that had been historically stored in different systems; the GNN, however, discovered strong connections based on the terms related to the workflow ("escalation path," "incident handling," "access provisioning"), thus disclosing the overlapping of the responsibilities across the teams.

- Improvement in Knowledge Retrieval: Users of the system noted that documents that were highly relevant to their work but that they "didn't know existed" were brought up by GNN-enhanced search. For example, engineers in the IT department who were looking for service-degradation protocols were shown the Operations incident logs that were relevant to them; thus, situational awareness was enhanced across the teams. Compliance-driven onboarding was supported by the HR teams getting connected to the IT security training documents.
- Semantic Grouping Usefulness: Topic communities produced by graph embeddings served intuitive clusters that were in line with the actual organizational workflows. The clusters like "Risk and Compliance," "Employee Lifecycle," and "System Deployment and Monitoring" appeared automatically, thus, enabling the teams to be quick in navigating through thematic knowledge spaces. These semantic groupings offered a more adequate framework than the usual folder-based or keyword-driven systems.

5.3. COMPARATIVE ANALYSIS

A comparative study against traditional information retrieval models and non-GNN graph approaches highlights the advantages and trade-offs of the proposed methodology.

1. Comparison with Traditional IR Models: TF-IDF, BM25, and cosine similarity methods remained effective when keywords appeared explicitly in text, but they struggled with conceptual mismatches or domain-specific variations in terminology. Classical NLP pipelines cluster documents based on surface-level similarity, missing deeper contextual relationships. The GNN-based approach significantly outperformed these methods by leveraging graph context—capturing entity co-occurrences, topic similarities, and multi-hop reasoning unavailable to traditional IR.
2. Comparison with Non-GNN Graph Approaches: Standard graph-based methods—such as purely similarity-based graphs or rule-driven knowledge graphs—lack adaptability. While they store connections, they do not *learn* relational patterns. Non-GNN graphs failed to infer missing links or generate enriched embeddings. In contrast, the GNN architecture expanded the graph's relational intelligence, enabling nuanced link prediction, topic propagation, and cross-domain inference.
    - Impact on Enterprise Workflows:The introduction of a GNN-based knowledge system improved several aspects of enterprise operations:
    - Reduced duplication of effort: Teams discovered existing documents instead of recreating similar content.
    - Enhanced cross-functional insight: Hidden relationships between departments surfaced.
    - Improved decision-making: Document clusters provided a clearer view of organizational knowledge flows. These changes directly enhanced productivity and alignment across business units.
3. Strengths and Limitations: Key strengths include scalability, relational reasoning, and the ability to unify heterogeneous content. However, the approach requires substantial computational resources for training and maintaining large graphs. Additionally, graph construction relies heavily on quality preprocessing and entity extraction; poor-quality metadata can propagate noise. Despite these limitations, the GNN model delivers capabilities that traditional enterprise search or IR-based systems cannot match.

6. CONCLUSION AND FUTURE SCOPE

The study introduced a detailed approach for utilizing Graph Neural Networks in enterprise knowledge mining, solving the problems that have existed for a long time, that is, the difficulty in handling large, fragmented, and semantically diverse document

ecosystems. Through the usage of semantic embeddings, heterogeneous graph construction, and advanced GNN architectures, the system demonstrated how companies can convert unstructured content into a connected, interpreted, and intelligence-driven knowledge space. The case study showed significant improvements in node classification, link prediction, clustering quality, and search relevance, thus revealing the capability of the system to uncover hidden relationships, decrease redundancy, and increase cross-departmental insights.

GNN technology for knowledge mapping is not only beneficial for information retrieval but also for other areas. By revealing relational patterns across documents, entities, and topics the system encourages the decision-making process to be more data-driven; thus, leaders will be able to see the dependencies and spot the risks that the traditional ways can't show. Furthermore, the relational insights that the system provides may be harnessed by automation use cases like smart recommender systems, workflow optimization, and compliance validation, while at the same time, better collaboration is promoted since interdepartmental knowledge becomes more visible and accessible.

However, a few limitations are acknowledged in this article despite the advancements made. Large-scale graphs of enterprises may experience data imbalance issues where certain departments or document types may be dominating the network, hence resulting in bias in model learning. Besides that, scalability also has its drawbacks; the training process involving millions of nodes and edges, despite models like GraphSAGE supporting inductive learning, is still very demanding in terms of computational resources. Another problem with GNN is interpretability since the complex architecture of GNN may hide the decision-making process. Although the use of attention mechanisms and graph explainability methods can help to some extent in this regard, the goal of completely interpretable enterprise AI is still a matter of ongoing research.

There are, however, several exciting potential directions that could further enterprise intelligence. The use of multimodal content integration, such as images, audio recordings, diagrams, and structured data, for example, could result in a single knowledge graph that encompasses all types of content. A combination of GNNs and LLM-based retrieval systems may create a hybrid reasoning scenario, where relational embeddings along with natural language understanding can be used for richer search experiences. The real-time updates for dynamic graphs would make it possible for the knowledge graph to be always up-to-date with the arrival of new documents and signals; thus, it would support time-sensitive operations. Lastly, the incorporation of the GNN-enhanced knowledge graph into enterprise chatbots and AI copilots can provide contextual intelligence right at people's everyday workflows; thus, employees would be able to query organizational knowledge in a conversational manner.

To sum up, a GNN-driven knowledge mapping is an essential move towards making the massive volume of content generated in enterprises comprehensible and usable, thereby setting the stage for more flexible, interconnected, and AI-enhanced organizations.

REFERENCES

[1] Han, Zhulin, and Jian Wang. "Knowledge enhanced graph inference network based entity-relation extraction and knowledge graph construction for industrial domain." *Frontiers of Engineering Management* 11.1 (2024): 143-158.

[2] Li, Pengjun, et al. "Survey and prospect for applying knowledge graph in enterprise risk management." *Computers, Materials and Continua* 78.3 (2024): 3825-3865.

[3] Tewari, Shishir, and Ashitosh Chitnis. "GRAPH-BASED MACHINE LEARNING FOR COMPLEX RELATIONSHIP DETECTION IN ENTERPRISE DATA." 2021,

[4] Ye, Zi, et al. "A comprehensive survey of graph neural networks for knowledge graphs." *IEEE Access* 10 (2022): 75729-75741.

[5] Xiong, Changri, et al. "Knowledge graph network-driven process reasoning for laser metal additive manufacturing based on relation mining." *Applied Intelligence* 54.22 (2024): 11472-11483.

[6] Du, Junliang, et al. "Graph Neural Network-Based Entity Extraction and Relationship Reasoning in Complex Knowledge Graphs." *2024 International Conference on Image Processing, Computer Vision and Machine Learning (ICICML)*. IEEE, 2024.

[7] Zheng, Xue, et al. "A knowledge graph method for hazardous chemical management: Ontology design and entity identification." *Neurocomputing* 430 (2021): 104-111.

[8] Qiu, Qinjun, et al. "Construction and application of a knowledge graph for iron deposits using text mining analytics and a deep learning algorithm." *Mathematical Geosciences* 55.3 (2023): 423-456.

[9] Yang, Bo, and Yi-ming Liao. "Research on enterprise risk knowledge graph based on multi-source data fusion." *Neural Computing and Applications* 34.4 (2022): 2569-2582.

[10] Liang, Zhuoqian, Ding Pan, and Yuan Deng. "Research on the knowledge association reasoning of financial reports based on a graph network." *Sustainability* 12.7 (2020): 2795.

[11] Song, Dezhao, et al. "Building and querying an enterprise knowledge graph." *IEEE Transactions on Services Computing* 12.3 (2017): 356-369.
[12] Guo, Zhiwei, and Heng Wang. "A deep graph neural network-based mechanism for social recommendations." *IEEE Transactions on Industrial Informatics* 17.4 (2020): 2776-2783.
[13] Lee, MyoungHoon, et al. "Technology opportunity discovery using deep learning-based text mining and a knowledge graph." *Technological Forecasting and Social Change* 180.121718 (2022).
[14] Chakraborty, Nilesh, et al. "Introduction to neural network-based question answering over knowledge graphs." *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 11.3 (2021): e1389.
[15] Wang, Hongwei, et al. "Knowledge-aware graph neural networks with label smoothness regularization for recommender systems." *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2019.