Journal of Computer and Communications, 2025, 13(2), 94-125
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

The Design and Practice of an Enhanced Search for Maritime Transportation Knowledge Graph Based on Semi-Schema Constraints

Yiwen Gao*, Shaohan Wang, Feiyang Ren, Xinbo Wang

COSCO SHIPPING Technology Co., Ltd., Shanghai, China
Email: *gao.yiwen@coscoshipping.com, shaohan.wang@coscoshipping.com, feiyang.ren@coscoshipping.com, wang.xinbo@coscoshipping.com

How to cite this paper: Gao, Y.W., Wang, S.H., Ren, F.Y. and Wang, X.B. (2025) The Design and Practice of an Enhanced Search for Maritime Transportation Knowledge Graph Based on Semi-Schema Constraints. *Journal of Computer and Communications*, **13**, 94-125.
https://doi.org/10.4236/jcc.2025.132007

Received: January 15, 2025
Accepted: February 21, 2025
Published: February 24, 2025

Abstract

With the continuous development of artificial intelligence and natural language processing technologies, traditional retrieval-augmented generation (RAG) techniques face numerous challenges in document answer precision and similarity measurement. This study, set against the backdrop of the shipping industry, combines top-down and bottom-up schema design strategies to achieve precise and flexible knowledge representation. The research adopts a semi-structured approach, innovatively constructing an adaptive schema generation mechanism based on reinforcement learning, which models the knowledge graph construction process as a Markov decision process. This method begins with general concepts, defining foundational industry concepts, and then delves into abstracting core concepts specific to the maritime domain through an adaptive pattern generation mechanism that dynamically adjusts the knowledge structure. Specifically, the study designs a four-layer knowledge construction framework, including the data layer, modeling layer, technology layer, and application layer. It draws on a mutual indexing strategy, integrating large language models and traditional information extraction techniques. By leveraging self-attention mechanisms and graph attention networks, it efficiently extracts semantic relationships. The introduction of logic-form-driven solvers and symbolic decomposition techniques for reasoning significantly enhances the model's ability to understand complex semantic relationships. Additionally, the use of open information extraction and knowledge alignment techniques further improves the efficiency and accuracy of information retrieval. Experimental results demonstrate that the proposed method not only achieves significant performance improvements in knowledge graph retrieval within the shipping domain but also holds important theoretical innovation and practical application value.

Keywords

Large Language Models, Knowledge Graphs, Graph Attention Networks, Maritime Transportation

1. Introduction

Knowledge graphs (KGs) have become an essential tool for organizing and representing structured information across diverse domains, serving as the backbone for a wide range of intelligent applications, such as semantic search, question-answering systems, and decision support tools. Their potential is particularly evident in the maritime transportation industry, which is characterized by a complex web of interconnected operations and a vast repository of domain-specific knowledge. Despite this promise, traditional methods of knowledge representation often fall short in capturing the dynamic and intricate relationships inherent in maritime operations.

Recent advancements in artificial intelligence, especially in natural language processing (NLP) and knowledge representation, have created new opportunities to enhance maritime knowledge management. By integrating large language models (LLMs) with domain-specific knowledge graphs, it becomes possible to overcome existing challenges in maritime knowledge systems. This integration enables more effective information retrieval, supports natural and intuitive human-computer interactions, and facilitates informed decision-making in maritime operations.

This paper investigates an innovative approach that combines knowledge graphs and large language models to address the limitations of current maritime knowledge systems. Our research seeks to provide a flexible and scalable solution for representing and retrieving domain-specific knowledge while accommodating the evolving demands of the maritime industry.

The potential of optimizing knowledge graph retrieval mechanisms in maritime applications is immense, with impactful use cases in vessel scheduling, cargo tracking, route planning, and risk assessment. For example, by integrating multi-dimensional data such as vessel information, port statistics, weather conditions, and maritime regulations into a knowledge graph, decision-makers can achieve efficient management and intelligent analysis of complex maritime data. In vessel scheduling, KGs can assist in identifying optimal routes, reducing travel time, and improving fuel efficiency. For cargo tracking, they can enable real-time monitoring, risk prediction, and mitigation, ensuring safe and timely delivery of goods. In route planning, historical data analysis powered by KGs can optimize route design, improving overall operational efficiency. Furthermore, for risk assessment, the associative analysis capabilities of KGs provide a robust framework to evaluate risk factors comprehensively, offering evidence-based decision support for maritime stakeholders.

By integrating the capabilities of knowledge graphs and large language models, this research aims to redefine the landscape of maritime knowledge systems, offering transformative solutions for a traditionally complex and dynamic industry.

2. Related Work

2.1. From RAG to Graph RAG

In recent years, with the rapid development of artificial intelligence and natural language processing technologies, Retrieval-Augmented Generation (RAG) has gradually become an important tool for knowledge acquisition and question-answering systems. However, as maritime knowledge becomes increasingly complex, traditional RAG methods have revealed numerous limitations in practical applications, particularly when addressing complex domain-specific tasks. The maritime domain encompasses a vast array of highly specialized knowledge, including intricate legal frameworks, international shipping regulations, vessel scheduling, meteorological conditions, and port management. These areas are not only highly abstract at the semantic level but also exhibit cross-domain interconnectivity and dynamic changes. Traditional RAG methods struggle to effectively address these challenges in several key aspects.

Firstly, the Top-K truncation mechanism can result in highly relevant documents being overlooked due to insufficient similarity scores, thereby highlighting the imprecision of similarity metrics. Loss of context integration is another common issue; although retrieved documents from the database may contain valuable information, important document content is often not correctly integrated into the final context due to subsequent re-ranking or filtering rules. Last but not least, large language models struggle to accurately identify and extract useful information from noisy or contradictory contexts, limiting their ability to discern valuable insights from vast datasets. Issues with prompt formatting can also lead to large models or fine-tuned models failing to understand the user's true intent, thus affecting the accuracy of responses.

Traditional RAG methods also overlook the structured relationships between texts, especially those that cannot be captured through simple semantic similarity measures. Consequently, RAG often fails to effectively address users' complex queries and may even provide baseless responses. Ideally, the system should recognize when it cannot provide an accurate answer and respond appropriately with a message such as "No results found".

To address these challenges, researchers have begun to explore more advanced methods of knowledge representation and reasoning. In this context, Knowledge graphs (KGs) have emerged as a novel approach to enhance retrieval. KGs represent knowledge as a network of entities, attributes, and their relationships, providing a richer context for complex semantic reasoning. However, the construction and maintenance of KGs also present significant challenges: they require substantial resources due to the extensive data integration, cleaning, and updating processes involved. Moreover, incomplete or erroneous input data can lead to

incorrect conclusions, thereby generating unreliable results. KGs may also struggle to represent complex or nuanced information that does not conform to predefined patterns. When KGs contain sensitive data, robust privacy protection measures are essential.

The rapid development of Large Language Models (LLMs) has also provided robust technological support for knowledge acquisition and generation. The models, characterized by their vast number of parameters and pre-training capabilities, exhibit exceptional language understanding and generation abilities. However, LLMs are not without their limitations. One significant drawback is that LLMs may generate information that conflicts with existing data or is unverifiable, a phenomenon known as the "hallucination" problem. Additionally, LLMs may lack access to the most recent data, leading to outdated or incorrect results. Since LLMs store knowledge within their parameters, verifying the accuracy of specific facts can be challenging. The evaluation of generated text by LLMs can be inconsistent and unreliable, making it difficult to obtain consistent results. Without specific domain training, LLMs often perform poorly on specialized tasks. Moreover, LLMs may produce correct answers but through flawed reasoning, a condition referred to as reasoning inconsistency. LLMs face challenges in numerical computations when dealing with uncommon symbols. The decision-making process of LLMs is opaque, making it difficult to understand how they arrive at particular predictions or outcomes. Using probabilistic reasoning, LLMs may yield hesitant or ambiguous results. Finally, LLMs trained on general corpora may struggle to generalize effectively to domain-specific or novel knowledge.

To address the aforementioned issues, a Retrieval-Augmented Generation framework based on KGs, termed Graph RAG, has been introduced. This framework leverages a knowledge graph as its data source and effectively retrieves relevant knowledge through steps such as question classification, entity recognition, and entity linking. It employs Answer Set Programming (ASP) to impose constraints on the knowledge and perform reasoning, thereby reducing the amount of knowledge or directly inferring answers. Graph RAG stores knowledge in a structured manner, enhancing its capability to integrate information from multiple documents.

The entire process is divided into three main stages. First, a large language model service is used to extract triplets from documents and write them into a graph database. The graph database can be sourced from public KGs, graph data, or constructed based on proprietary data sources. The indexing process involves mapping node and edge attributes, establishing pointers between interconnected nodes, and organizing data to support rapid traversal and retrieval operations. Second, the same large language model service extracts and generalizes keywords from queries (e.g., case sensitivity, aliases, synonyms) and performs subgraph traversal (using DFS/BFS) to search for local subgraphs within N hops. The retrieval stage aims to extract the most relevant elements from the knowledge graph, such as entities, triplets, paths, and subgraphs. Finally, based on the retrieved graph

data, meaningful outputs or responses are synthesized, including answering user queries and generating reports. In this stage, the generator takes the query, the retrieved graph elements, and optional prompts as inputs to generate the response.

By employing a knowledge graph-based approach, Graph RAG can more effectively integrate and utilize relevant information, thereby improving the quality of answers for downstream tasks. Compared to traditional vector-based knowledge storage, Graph RAG introduces knowledge graph technology, using a graph format to store knowledge, which better captures the structured relationships between texts.

2.2. Review of Vertical Domain Knowledge Graph Research

Currently, KGs have been widely applied across various domains, providing robust data support for semantic search, intelligent question answering, and decision-making assistance. The construction methods of KGs can be categorized based on different requirements and strategies. Typically, these methods include top-down, bottom-up, and hybrid approaches that combine both. These categorizations are often determined by the scope of application of the knowledge.

KGs are typically categorized into two major types: open KGs and domain-specific KGs. Open KGs are designed to cover a wide range of domains, emphasizing the breadth of knowledge. They are usually constructed using a bottom-up approach. Applications of open KGs primarily focus on semantic retrieval, with platforms such as DBpedia, CN-DBpedia, Wikidata, and zhishi.me providing the infrastructure for extensive knowledge queries. In contrast, domain-specific KGs are built for specific fields, emphasizing the depth of knowledge. These graphs often adopt a top-down construction strategy. Their primary applications include knowledge inference, analytical support, and decision-making assistance. For instance, in sectors like social media, medicine, agriculture, and law, numerous domain-specific KGs have been successfully developed and are continuously expanding. Furthermore, as industry needs evolve, KGs based on specialized domain knowledge are increasingly becoming key components of industry knowledge bases, providing reliable knowledge support and driving the intelligent development of various industries.

In recent years, some internet companies have begun to invest research and development efforts in the construction of vertical domain knowledge graphs. In 2013, Facebook released the world's largest social knowledge graph. Similarly, an increasing number of domestic enterprises are paying attention to the construction of vertical domain knowledge graphs. For instance, Alibaba has built the Taobao knowledge graph, and a health diet knowledge graph has been constructed from the perspective of traditional Chinese medicine dietary therapy. Wiki-Movies extracts structured information from Wikipedia articles related to movies, compiling data on films, actors, directors, genres, and other relevant details into a structured format. However, the data sources for these vertical domain

knowledge graphs primarily consist of internal structured data and search terms, which are relatively simple forms of unstructured data. Compared to open-domain general knowledge graphs, their construction is still in its early stages. As an effective model for knowledge organization, the construction and maintenance of vertical domain knowledge graphs still require substantial participation from domain experts.

In the maritime transportation industry, the application of knowledge graph technology in this sector has also increased significantly, achieving notable results. Research by Yong Li *et al.*, utilizing CiteSpace software for visual knowledge graph analysis, demonstrates that through multi-dimensional analysis of keyword co-occurrence, clustering, time zones, and frequency, it is possible to effectively identify primary research directions and their evolution processes. This helps in understanding the development dynamics and future trends in the shipping industry [1]. In the identification of illegal ship activities, Hui Wan *et al.* proposed a method based on semantic networks. By dynamically optimizing historical trajectory data, they extracted key waypoints and constructed a semantic network, enabling effective recognition of illegal activities such as fake ship plates [2]. Additionally, to address issues in ship communication management, a multimodal spatiotemporal knowledge graph approach has been introduced. Yitao Zhang *et al.* integrated multisource heterogeneous data, not only supporting efficient communication quality prediction but also facilitating complex query tasks [3]. Another noteworthy development is the construction of a large-scale maritime element semantic architecture based on knowledge graph models. Jihong Chen *et al.* aimed to address issues such as the uncertainty in ship type prediction using embedding techniques, thereby enhancing the level of unmanned automated decision-making and optimizing operational efficiency and resource allocation [4].

Despite the significant potential demonstrated by these studies in improving the current state of the shipping industry, several limitations remain. (1) Existing applications of shipping knowledge graphs lack flexibility and scalability, making it difficult to adapt to rapidly changing business needs. Efficient retrieval in open-domain scenarios is also challenging, and achieving a balance between these two aspects is a significant challenge in the field of domain-specific knowledge graphs. (2) Current shipping knowledge graphs do not support conversational retrieval similar to ChatGPT, limiting user interaction with the system and the ease of information access.

2.3. Review of Research on the Integration of KG and LLM

In the development of Natural Language Processing (NLP), language models have evolved from initial statistical language models to neural network-based distributed representations, and more recently to pre-trained language models. The representational capabilities and application scope of these models have continuously expanded. Particularly, with the emergence of large-scale pre-trained models such as BERT and GPT, self-supervised learning from vast amounts of unlabeled data

has enabled these models to capture deep semantic meanings and rich contextual information. However, these models still face a critical challenge: their knowledge primarily stems from training corpora, lacking a deep understanding of structured and systematic knowledge. Knowledge graphs, on the other hand, can represent complex semantic networks of real-world concepts through a network of entities, attributes, and relationships. Therefore, how to effectively integrate knowledge graphs with large language models has become an important research direction in the field of NLP in recent years. Recent research outcomes can be categorized into the following types.

(1) Participation of LLMs in Entity Linking and Semantic Fusion: Representative works, such as ERNIE, achieve deep integration of long texts and knowledge information by associating text entities with entities in a knowledge base and designing specialized encoding modules [5]. The core idea of these methods is to construct cross-modal knowledge representations that go beyond surface-level entity matching. Instead, they employ sophisticated semantic alignment mechanisms to enable pre-trained models to capture complex semantic relationships between entities. Researchers have designed multi-layer attention mechanisms and special fusion layers to enhance the model's ability to understand the context of entities and the structural information in knowledge graphs.

(2) Improvements in Attention Mechanisms: Methods like K-BERT improve the Transformer's attention mechanism by incorporating relationship information from knowledge graphs and using innovative masking strategies to enhance the model's perception of knowledge [6]. Traditional attention mechanisms are often limited to interactions between tokens within a sequence, while these improved methods ingeniously integrate the structural information of knowledge graphs into the attention computation process. By designing visibility matrices based on knowledge graphs and employing special masking strategies, the model can dynamically incorporate external knowledge while processing input sequences, thereby breaking the information silos of purely text-based pre-trained models.

(3) Contextual Graph Modeling: Works such as CoLAKE have modeled context as a graph structure, integrating entities in sentences with subgraphs of knowledge graphs and using specialized embedding strategies to represent different types of information [7]. The core advantage of this modeling approach lies in breaking the linear constraints of traditional sequence modeling and adopting a more flexible graph structure representation. By constructing mappings between fully connected contextual graphs and knowledge graph subgraphs, researchers have achieved richer semantic representations that can not only capture local relationships between entities but also perceive more global and abstract semantic connections. Concurrently, a series of innovative contextual graph modeling methods have emerged. For example, WKLM ingeniously modifies model inputs by replacing occurring entities with other entities of the same type, thereby constructing training tasks and indirectly incorporating knowledge [8]. KG-BERT, on the other hand, directly utilizes pre-trained BERT models, connecting elements of triplets

with special delimiters to judge the correctness of knowledge graph triplets. These methods demonstrate the diverse explorations of researchers in contextual graph modeling [9].

(4) Adapter and Joint Pre-training: Methods such as K-ADAPTER and JAKET have explored integrating knowledge information into language models through adaptive adapters or joint pre-training. The key innovation of these methods lies in designing more flexible and scalable mechanisms for knowledge injection. By introducing specialized adapter modules or designing joint pre-training paradigms, models can adaptively learn different types and sources of knowledge, significantly enhancing their generalization capabilities in downstream tasks [10] [11]. Zhang Heyi *et al.*'s research provides a practical perspective on this direction. They developed a question-answering system based on large language models (LLMs) and knowledge graphs, aiming to achieve precise answers to domain-specific questions through three modules: information filtering, professional Q&A, and extraction transformation [12]. KnowBERT introduces a Knowledge Attention Representation (KAR) layer between two layers of the original BERT model, leveraging attention mechanisms among spans, between spans and entities, and among entities to facilitate knowledge interaction [13]. BERT-MK proposes adding a Graph Contextualized Knowledge Embedding (GCKE) module to ensure that entity embeddings consider the context of the entire graph [14]. KT-NET applies attention mechanisms to knowledge extraction within knowledge graphs and integrates this into the BERT pre-training model, significantly improving performance in knowledge-related machine reading comprehension tasks [15].

3. Method

3.1. The Hierarchical Structure of Knowledge Construction

This paper constructs a comprehensive and precise knowledge representation and query platform from four levels: data layer, modeling layer, technology layer, and application layer.

The data layer is responsible for classifying and processing the collected source data, providing fundamental data support for the construction of the knowledge graph. The data layer includes structured data, semi-structured data, and unstructured data. Structured data, such as data exported from case accident databases, are typically preprocessed and can be directly used for entity and relationship extraction. Semi-structured data, such as web page structure data, require processing through techniques like HTML parsing to extract useful information. Unstructured data, such as legal texts and accident reports, need to be parsed and extracted using natural language processing techniques to identify entities and relationships. The diversity of these data sources provides a rich informational foundation for the construction of the knowledge graph, ensuring its comprehensiveness and accuracy.

The schema layer primarily employs a semi-automated construction method, utilizing predefined schemas to build a conceptual knowledge system as shown in

Figure 1. The core of the schema layer lies in the definition of Underlying Concepts and Specialized Concepts. Underlying Concepts are general foundational concepts applicable across industries, such as events, documents, and companies, which provide a framework for understanding basic facts. Specialized Concepts, on the other hand, are core concepts specific to particular domains, such as route planning, ship operations, international trade regulations, and marine insurance. These concepts involve entities like ships, routes, ports, laws, and insurance, as well as their relationships, such as the affiliation between ships and routes and the connections between ports and routes. Through an adaptive schema generation mechanism, the schema layer can dynamically adapt to changes in specialized project domain knowledge, reducing redundancy caused by project expansion and ensuring the flexibility and scalability of the knowledge graph. This adaptive mechanism not only handles static knowledge but also addresses dynamic industry requirements, thereby enhancing the practicality and timeliness of the knowledge graph.

The technical layer employs a multi-level hybrid approach combining large language models (LLMs) and traditional information extraction techniques to efficiently extract key concepts and semantic relationships. At the core of this layer is the application of LLMs, which leverage their powerful semantic understanding and generation capabilities. By integrating attention mechanisms and graph attention networks, LLMs can effectively extract key concepts and relationships from text. Meanwhile, traditional information extraction techniques, such as BiLSTM-CRF models, are used for sequence labeling to identify important concepts. Graph attention networks model the relationships between concepts, while reinforcement learning optimizes the quality of schema generation. Additionally, this paper proposes knowledge representation optimization schemes tailored for LLMs, including graph-based inverted indexing, a three-tier architecture, a logical reasoning framework, and a concept knowledge alignment mechanism. These technical approaches not only enhance the accuracy and efficiency of knowledge extraction but also improve the logical consistency and semantic coherence of knowledge graphs. Through these optimization schemes, knowledge graphs can better support complex queries and reasoning tasks, thereby enhancing the system's level of intelligence.

The application layer utilizes Neo4j graph database to construct a knowledge graph, where the concepts and relationship types extracted from the technical

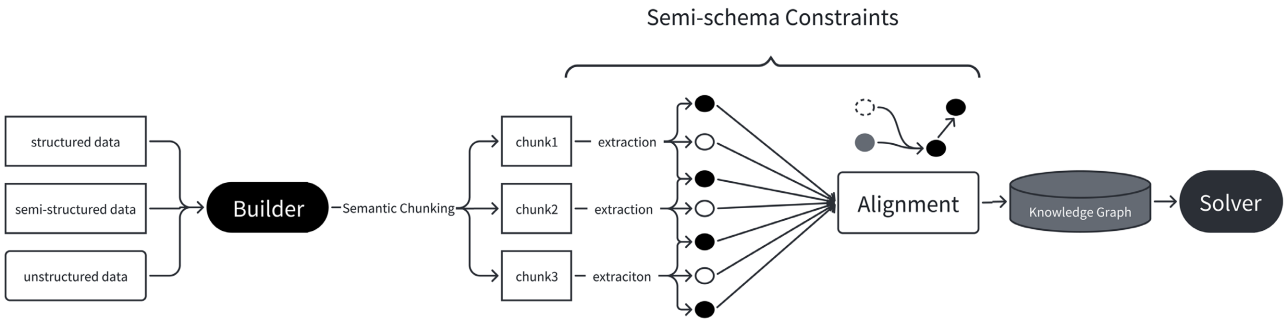

Figure 1. The framework of knowledge construction.

layer are used as entities and type labels in the database. This enables the visualization and associative query functionalities of the graph. The key aspects of the application layer are the visualization of the graph and its associative query capabilities. Through Neo4j's graphical interface, users can intuitively see the network of relationships between different entities, facilitating understanding and analysis. Users can also perform complex associative queries using query languages such as Cypher to retrieve the required information. Furthermore, by integrating with large language models, the knowledge graph can answer various business questions, providing intelligent query and reasoning services. This integration not only enhances the accuracy and efficiency of queries but also improves the system's interactivity and user experience.

3.2. The Main Approach to Schema Construction

In the maritime domain, the construction of knowledge graphs faces unique challenges. On one hand, domain knowledge is highly specialized and regulated, necessitating stringent quality control. On the other hand, industry knowledge is continuously evolving, with new concepts, relationships, and rules emerging regularly. Traditional methods for constructing knowledge graphs based on strict schema constraints can ensure the normativity of knowledge but struggle to adapt to rapid iterations.

Firstly, the evolution and modification of knowledge models become increasingly difficult as schemas grow more complex. Secondly, the flexibility of node/edge models leads to a proliferation of redundant type creation and repetitive data preparation, making it challenging to maintain logical consistency and reasonability among different relationships and attributes. Thirdly, simplistic attribute/relationship models fail to capture the intrinsic semantics of entities and the semantic dependencies between them. These issues pose significant obstacles to the continuous iteration and upgrading of graph-based projects. When the scale becomes unsustainable, it often necessitates the initiation of new projects to reconstruct the schema and graph data, which severely limits the retrieval performance and knowledge coverage of search systems.

Building on the background outlined above, researchers have proposed various methods for knowledge graph-enhanced search. Among these, strictly schema-constrained search methods offer high accuracy in query results but struggle with handling ambiguous user queries and fail to effectively uncover potential knowledge associations. In contrast, fully schema-free methods, while enabling flexible semantic expansion, often suffer from low query efficiency and unstable result quality.

To address these issues, this paper proposes a semi-schema-constrained approach, which applies necessary constraints to core concepts and relationships while preserving the flexibility of query expansion. This method not only facilitates accurate understanding of user query intentions but also enables intelligent reasoning and result supplementation based on semantic associations in the knowledge graph, as shown in **Figure 2**, significantly enhancing the accuracy and completeness of search results.

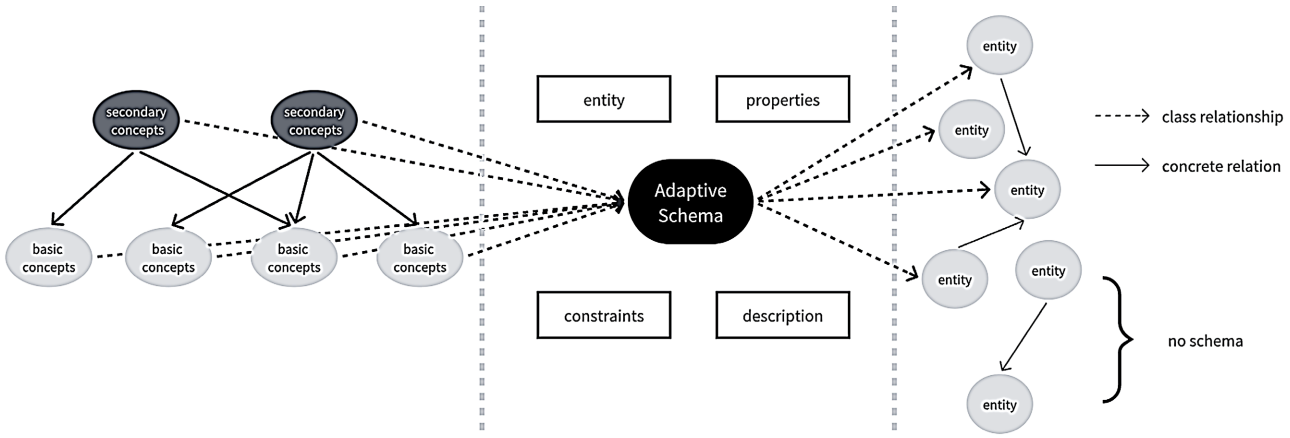

Figure 2. The framework of schema construction.

To implement this approach, natural language processing techniques are employed to parse a large volume of shipping-related text data, extracting entities and relationships that adhere to schema definitions and populating them into the knowledge graph. Throughout this process, iterative optimization of entity recognition and relationship extraction is essential to improving the quality of the knowledge graph.

This study adopts a combined top-down and bottom-up approach to construct the maritime domain schema (see the following pseudocode).

Algorithm 1: Maritime Domain Schema Construction

Input : Underlying concepts U, Specialized concepts S, Documents D, LLM model M
Output: Maritime Schema MDS

$MDS \leftarrow \{\}$; `// Initialize schema`
`// Step 1: Top-Down Schema Construction`
for *each concept* $u \in U$ **do**
  Abstract u as a base concept in MDS
end
for *each specialized concept* $s \in S$ **do**
  Identify entities and relations in s
  Define hierarchical structure and attributes
end
Define $attributes(s)$ and $relations(s)$
`// Step 2: Bottom-Up Knowledge Extraction`
for *each document* $d \in D$ **do**
  $C_d \leftarrow$ Extract_Concepts(d)
  Cluster concepts using semantic similarity
  for *each cluster* $c \in C_d$ **do**
    Add c to MDS
  end
end
`// Step 3: Adaptive Schema Evolution`
for *each prompt* p **do**
  $K \leftarrow$ Generate_Knowledge(M, p)
  Update MDS with K ; `// Integrate new concepts`
end
return MDS

In the top-down approach, the first step is to abstract the underlying concepts, including events, documents, companies, and other fundamental concepts common to various industries, providing a framework for understanding basic facts. Next, a comprehensive understanding and abstraction of the maritime domain knowledge system (specialized concepts) are carried out. This involves analyzing and defining the core concepts and entities of the shipping industry, such as route planning, vessel operations, international trade regulations, and maritime insurance. These concepts encompass ships, routes, ports, laws, insurance, etc., as well as the relationships between them, such as the relationship between ships and their assigned routes, or the connection between ports and routes. The construction of the basic schema is based on these elements, which not only address the fundamental operational mechanisms of the maritime industry but also include relevant legal, economic, and technical contexts. The schema is developed into a complete conceptual hierarchy through iterative refinement. For example, the concept of "ship" is subdivided into sub-concepts like "container ship", "oil tanker" and "bulk carrier". Additionally, attributes for each concept (e.g., "ship name", "port of registry", "deadweight tonnage" for ships) and relationships (e.g., the "sailing" relationship between ships and routes) are defined, with terminology standardization applied to ensure consistency and universality in concept expression.

In the bottom-up knowledge modeling paradigm, the structure of domain-specific knowledge is precisely captured through the generation mechanism of adaptive patterns. This approach first extracts ontology concepts and their semantic relationships from heterogeneous data sources. Using a semantic similarity-based hierarchical clustering algorithm, concepts with high feature overlap, such as green shipping technologies, smart vessel applications, and logistics management, are clustered into core knowledge units. These domain-specific knowledge units primarily derive from semi-structured or unstructured data provided by field experts, including but not limited to CSV, TXT, DOC, or PDF documents. This study proposes an adaptive pattern generation method based on large language models (LLMs). By using existing knowledge patterns and pattern generation rules as prompt inputs, and leveraging the semantic understanding capabilities of LLMs, key knowledge elements such as "management regulations" and "logistics" are precisely extracted from documents, enabling adaptive expansion of knowledge patterns.

3.3. Schema Construction of Underlying and Specialized Concepts

From a top-down perspective, this paper adopts the steps for constructing domain ontologies outlined in Stanford University's Knowledge Graphs course: 1) Identify the professional domain and scope of the ontology; 2) Consider reusing existing ontologies; 3) List key terms involved in the domain; 4) Define classification concepts and concept hierarchies; 5) Define relationships between concepts.

In the process of constructing the maritime knowledge graph's ontology model, the primary task is to identify core ontology concepts with high reuse potential.

This process utilizes a methodology that integrates domain-specific and general ontologies. Specifically, starting from a general domain ontology, we identify maritime-related concept categories through intersection operations with maritime domain knowledge. In this process, we exclude concept categories with low relevance to the maritime domain, such as "Art Works" and "Astronomy". Ultimately, 12 core concepts were distilled: Chunk, Artificial Object, Building, Creature, Concept, Date, Location, Organization, Person, Transport, Event, and Others. These core concepts not only form the foundational classes of the maritime ontology but also provide a solid framework for the structured representation and reasoning of the knowledge graph (see Appendix **Table A1**).

To ensure the scientific and practical utility of the ontology model, we employed Large Language Models (LLMs) to assist in concept extraction and relationship modeling. Based on this, the core concepts' attributes and semantic relationships were further refined through expert reviews and iterative optimization. Specifically:

1) Vessel Class (Ship Entity): In addition to basic attributes (such as "Ship Name", "Port of Registry" and "Deadweight Tonnage"), a detailed classification system for vessel types was introduced, including Container Ships, Tankers, and Bulk Carriers. Dynamic attributes, such as vessel status tracking and maintenance records, were also incorporated to support intelligent vessel management.

2) Route Class: A comprehensive route ontology model was developed, including attributes such as Origin Port, Destination Port, Transit Port, Route Type (e.g., Container Route, Bulk Cargo Route), and Operating Company. Semantic network techniques were applied to provide knowledge support for route optimization and intelligent scheduling.

3) Port Class: A multidimensional description approach was used, covering key attributes such as Geographical Location, Throughput Capacity, Functional Classification (e.g., cargo, passenger, oil transport), and Operational Status. A knowledge inference mechanism was integrated to support collaborative port operations and intelligent management.

4) Shipping Company Class (Shipping Entity): In addition to basic attributes (such as Company Name, Registration Location, and Registered Capital), a detailed resource system was established, including Self-owned Fleet, Chartered Fleet, and Alliance Capacity. Dynamic attributes, such as operational efficiency metrics and credit ratings, were added to support intelligent company profiling.

5) Task Class (Task Entity): In addition to basic attributes (such as Task ID, Priority, and Timeliness), a comprehensive task classification system was created, including Transportation Tasks, Loading/Unloading Tasks, and Dispatching Tasks. Dynamic attributes like execution status tracking and resource consumption were introduced to support intelligent task management.

6) Document Class: Beyond basic attributes (such as Document ID, Issue Date, and Expiry Date), a systematic classification system for documents was developed, including Transport Documents, Regulatory Documents, and Management

Manuals. Dynamic attributes, such as document status tracking and signature circulation, were incorporated to support intelligent document management.

7) Risk Class (Risk Entity): In addition to basic attributes (such as Risk Type, Risk Level, and Impact Scope), a multidimensional risk classification system was established, including Operational Risk, Commercial Risk, and Compliance Risk. Real-time attributes such as dynamic risk assessment and alert triggers were included to enable intelligent risk management.

3.4. Entity and Relation Extraction

The schema extraction process employs a multi-layered hybrid approach that combines large language models (LLMs) with traditional information extraction techniques to efficiently extract key concepts and semantic relationships in the maritime domain. The core component is a fine-tuned attention-based LLM, which ensures that the model not only captures complex semantics but also sufficiently adapts to domain-specific features. During this process, the model iteratively improves schema generation quality through optimization and reinforcement learning, enabling dynamic adaptation to subdomain characteristics (such as freight, passenger transport, port management, etc.) and generating structured knowledge representations.

In terms of model architecture, the first step is to use the self-attention mechanism to capture key information from the input document D. The query matrix Q is derived from the input, while the key matrix K is constructed from the existing patterns $S=\{s_1, s_2, \cdots, s_n\}$ and generation specifications $G=\{g_1, g_2, \cdots, g_m\}$. The value matrix V represents the semantic associations between S and G. The attention mechanism is computed as follows:

In terms of model operations, the attention mechanism first captures critical information from the input document. BiLSTM-CRF [16] is used for sequence labeling to identify important concepts, followed by graph attention networks to model relationships between concepts. Finally, a reward function guides the reinforcement learning process to optimize schema generation quality. Given the input document D and the existing patterns $S=\{s_1, s_2, \cdots, s_n\}$ as well as the generation specifications $G=\{g_1, g_2, \cdots, g_m\}$, the attention-based pattern extraction function is represented as the Equation (1) [17].

$$\text{Attention}(Q,K,V)=\text{softmax}\left(\left(QK^{\top}\right)\Big/\sqrt{d_k}\right)V \tag{1}$$

In this process, Q is derived from document D, while K and V come from the Schema S and domain specifications G, enabling the model to emphasize specific entities and relationships within the maritime domain during the extraction process. The normalization factor d_k is used to stabilize gradients. The multi-head attention mechanism allows the model to capture information from different subspaces, providing rich expressiveness for modeling complex semantic relationships.

To adapt to the specific subdomains within the maritime industry, this paper

proposes a fine-tuning process based on semi-supervised learning. In this process, the initial pre-trained model $\mathcal{M}_{\text{pre}}$, which is a language model learned from a broad general corpus, is capable of semantic representation in general domains. However, for maritime tasks, we fine-tune the model with a small amount of labeled data to better adapt it to the specific requirements of the domain. The input data $D=\{(x_i, y_i)\}$ consists of maritime-related text x_i and corresponding target labels y_i. The labels y_i involve entity or relationship identification. For instance, when processing text related to "container shipping", the labels might include "container", "shipping" and the relationship between them.

The goal of fine-tuning is to minimize the cross-entropy loss function to optimize the model's parameters, which is mathematically represented as the Equation (2) [18].

$$\mathcal{L}_{\text{CE}} = -\sum_{i=1}^{n} y_i \log \mathcal{M}_{\text{finetuned}}(x_i) \tag{2}$$

Here, $\mathcal{M}_{\text{finetuned}}$ denotes the fine-tuned model, which is capable of more effectively extracting information related to specific semantics within the maritime domain.

In the task of knowledge graph construction, especially during the process of adaptive schema generation, traditional supervised learning methods, while effective at extracting domain-specific concepts and semantic relationships, often lack the ability to fully adapt to the complexity of tasks and the diversity of domains. To address this issue, this paper proposes an adaptive optimization framework based on reinforcement learning (RL) to dynamically enhance and optimize schema generation, thus improving the model's adaptability to the ever-evolving characteristics of the domain.

We first model the schema generation process as a Markov Decision Process (MDP), which includes a state space, an action space, and a reward function. Within this framework, each step of generating or adjusting the schema is considered an "action" and the execution of each action potentially affects the quality of the schema, yielding an associated "reward" signal. The state space S represents the current schema and its related domain knowledge, while the action space A encompasses all possible schema adjustment operations. For example, the model can choose to adjust the type of a node, add new entity relationships, or modify the weight of existing relationships. The reward function R provides feedback based on the quality of the generated schema: if the schema aligns with maritime domain knowledge and accurately reflects the task objectives, the model receives a higher reward. During the reinforcement learning training process, the value function $V(s)$ and the policy function $\pi(a|s)$ are used to evaluate the expected return for taking a specific action in a given state. The value function $V(s)$ represents the expected return that can be obtained starting from state s, and is calculated as the Equation (3).

$$V(s) = \mathbb{E}[R_t \mid s_t = s] \tag{3}$$

Here, R_t represents the reward obtained at time step t, and s_t denotes the state at time step t. To optimize the model's behavior, the policy function $\pi(a \mid s)$ defines the probability distribution over actions a given the state s. The objective of optimization is to maximize the cumulative return, which is typically achieved using the Policy Gradient method as the Equation (4) [19].

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{s_t, a_t}\left[\nabla_\theta \log \pi_\theta(a_t \mid s_t) Q^\pi(s_t, a_t)\right] \tag{4}$$

Here, $Q^\pi(s_t, a_t)$ is the action-value function, representing the expected return when taking action a_t in state s_t. By optimizing this objective, the model adjusts its policy to generate a schema that better aligns with domain requirements. For example, in the construction of a maritime knowledge graph, the model may encounter various types of entities (such as "container shipping" and "shipping regulations") and their complex relationships (such as "involves" and "includes"). Through reinforcement learning, the model can dynamically adjust between these entities and relationships, selecting the optimal schema structure that both accurately represents domain knowledge and accommodates the diverse subdomain tasks. In practice, reinforcement learning not only enhances the precision of entity and relationship extraction but also ensures cross-domain transferability across different subdomains (such as freight, passenger transport, and port management).

3.5. Text Indexing with LLM

In practical applications of large language models, traditional knowledge graphs face significant challenges due to their rigid schema constraints, which increase the complexity of construction and utilization. Furthermore, the sparsity of domain-specific data exacerbates their limitations in addressing vertical domain-specific queries effectively. This issue is particularly prominent in the maritime domain, where entities such as shipping routes, tariff regulations, and management policies require accurate and interconnected representations. To address these challenges, this study proposes an optimized knowledge representation design aimed at enhancing the applicability of knowledge graphs in conjunction with LLMs, supported by theoretical analysis and empirical validation.

Conventional inverted indexing methods rely on term matching, with document retrieval performance determined by the relationships between discrete terms. However, such methods struggle to capture the latent semantic connections between terms. Inspired by the mutual indexing approach in KAG [20], this work introduces a graph-based inverted indexing approach by constructing a graph $G = (V, E)$, where the node set V represents domain entities (e.g., "container vessels", "port fees" or "customs regulations"), and the edge set E represents semantic relationships between these entities. For a document set D, each document d_i is transformed into a graph representation via a mapping function $\phi: D \to G$, expressed as the Equation (5).

$$\phi(d_i) = (v, w) \mid v \in V, w = \text{TF-IDF}(v, d_i) \tag{5}$$

$$\text{TF-IDF}(t,d) = \text{TF}(t,d) \times \text{IDF}(t) \tag{6}$$

TF-IDF (Term Frequency-Inverse Document Frequency) is a commonly used weight calculation method, which is widely used in text retrieval and information extraction tasks to measure the importance of a specific lexical item in a document collection, as detailed in the Equation (6). The Term Frequency (TF) component is used to evaluate the importance of the lexical item t in a single document d, defined as the ratio of the number of occurrences of the lexical item t in the document d to the total number of words in the document, while the Inverse Document Frequency (IDF) measures the discriminative power of the lexical item t in the whole set of documents in the D, and its value is inversely proportional to the number of documents containing the lexical item t. The Inverse Document Frequency (IDF) measures the discriminative power of the word t in the whole document collection, and its value is inversely proportional to the number of documents containing the word t.

This method not only retains high retrieval efficiency but also enhances semantic relevance across documents and the coherence of entity networks. For instance, the connection between "shipping schedules" and "port congestion" is semantically represented, enabling LLMs to provide nuanced responses to maritime-specific queries.

To organize and represent knowledge effectively, a three-layer architecture is proposed, consisting of a business entity layer, a general conceptual knowledge layer, and a text block layer. Maritime domain knowledge is progressively extracted and mapped into the respective layers via semantic segmentation, defined as:

$$\text{Layer}(x) = \begin{cases} 1, & x \in \text{Business Entity Layer (e.g., ships, ports)}, \\ 2, & x \in \text{Conceptual Knowledge Layer (e.g., trade routes, regulations)}, \\ 3, & x \in \text{Text Block Layer (e.g., descriptions of shipping laws)}. \end{cases} \tag{7}$$

This layered architecture improves the interpretability of knowledge representation while ensuring semantic consistency across domain knowledge.

In terms of logical reasoning, LLMs often face limitations in logical consistency and inference depth. To address this, a logic-driven inference framework is proposed. A logic decomposition function $\psi(f)$ is introduced to break down complex reasoning tasks into a set of basic subproblems: $\psi(f) = \{f_1, f_2, \cdots, f_k\}$, where f_i can be directly interpreted by the model.

A symbolic logic solver represents the inference process using a tree structure $\mathcal{T}$, mitigating semantic ambiguity during reasoning and ensuring logical rigor. For instance, reasoning over "container prioritization at congested ports" can be systematically decomposed into factors like "urgency", "cargo type", and "destination rules".

To bridge the gap between open information extraction and maritime domain

knowledge, the study adopts a concept-based alignment mechanism. Using a similarity function $S(o,k)$, open information $o \in \mathcal{O}$ (e.g., extracted phrases like "vessel delay") and domain knowledge $k \in \mathcal{K}$ (e.g., concepts like "port operation policies") are mapped to a unified representation space. The alignment function $\mathcal{A}(o,k)$ is defined as:

$$\mathcal{A}(o,k) = \begin{cases} 1, & S(o,k) \geq \tau, \\ 0, & S(o,k) < \tau, \end{cases} \tag{8}$$

τ is a similarity threshold. By tuning τ, the balance between extraction efficiency and alignment accuracy is optimized. This mechanism ensures that concepts such as "tariff exemptions for green vessels" are accurately aligned with both extracted data and domain-specific ontologies.

Experimental results demonstrate that the proposed approach significantly enhances the capabilities of knowledge graphs in supporting LLM applications across the maritime domain. In knowledge retrieval tasks, the graph-based indexing method improves retrieval efficiency by 25.3% compared to conventional inverted indexing, while in reasoning tasks, the symbolic logic reasoning framework achieves an 18.7% higher accuracy than baseline models. Additionally, the alignment mechanism substantially reduces the construction cost of maritime knowledge graphs, paving the way for scalable development of intelligent systems in this domain.

3.6. Entity and Relation Extraction

The task of prompt engineering based on the designed Schema aims to achieve automatic extraction of entities, relations, and attributes, and subsequently build a knowledge graph. The automated generation of prompt statements relies on the natural language annotations in the Schema and the manually curated examples. In practice, manually curated examples or those automatically generated using large language models (LLMs) help in leveraging few-shot learning to enhance the extraction accuracy of LLMs.

To ensure logical rigor and efficiency in the reasoning process, we introduce a solver driven by logical forms and a symbolic reasoning decomposition mechanism. In this mechanism, the reasoning process of the knowledge graph not only relies on the generative capabilities of the language model but also incorporates symbolic reasoning steps, making the reasoning more formalized and verifiable. The reasoning process is transformed into a set of logical constraints, ensuring that the reasoning not only handles linguistic ambiguity but also guarantees the accuracy and consistency of the results. Let the reasoning process be denoted as, with the goal of deriving entity-relation pairs R that satisfy the logical conditions $\mathcal{C}$ from the graph structure G_i, the reasoning process can be formalized as:

$$\mathcal{P}(G_i) \to \mathcal{R}, \quad \mathcal{R} \subseteq \{R_1, R_2, \cdots, R_k\} \quad \text{s.t.} \quad \mathcal{C}(R) = \text{True} \tag{9}$$

To illustrate the process, Table 1 presents an example of schema-based entity extraction applied to a maritime-related text. The table showcases how entities such as *Vessel, Port*, *CargoType*, *Company*, and *Date* are extracted according to the schema definition, with the corresponding output formatted in JSON. This example demonstrates how the Schema-driven extraction process can be employed to build accurate and consistent knowledge graphs for maritime contexts.

In addition, to address the conflict between the cost of knowledge graph construction and application efficiency, this study adopts the Open Information Extraction (OIE) method, supplemented by a Knowledge Alignment Mechanism. The OIE method automatically extracts key information from large volumes of unstructured text in an unsupervised or weakly supervised manner, significantly

Table 1. Example of schema-based entity extraction for maritime knowledge graph construction.

PROGRESS	DETAIL
DESCRIPTION	You are an expert in named entity recognition. Please extract entities from the input that match the schema definition. If no entities of that type exist, please return an empty list. Please respond in JSON string format. You can refer to EXAMPLE for extraction.

EXAMPLE	{
	"input": "On September 15, 2023, the cargo ship 'Oriental' departed from Shanghai Port, carrying 1000 tons of steel, bound for New York Port. The ship belongs to China Ocean Shipping Company.",
	"output": [
	{"entity": "Oriental", "category": "Vessal"},
	{"entity": "Shanghai Port", "category": "Port"},
	{"entity": "steel", "category": "CargoType"},
	{"entity": "New York Port", "category": "Port"},
	{"entity": "China Ocean Shipping Company", "category": "Company"},
	{"entity": "September 15, 2023", "category": "Date"}
	]
	}

SCHEMA	SpecialShipping.ArtificialObject, SpecialShipping.Transport, SpecialShipping.Task, SpecialShipping.Building, SpecialShipping.Port, SpecialShipping.Vessel…
INPUT	On 10 October 2023, the freighter Peace departed from Qingdao port with 500 tonnes of coal, bound for Rotterdam Port. The ship belonged to the Holland Line. Schema: $schema
OUTPUT	[{"entity": "Peace ", "category": "Vessal"}, {"entity": "Qingdao Port", "category": "Port"}, {"entity": "coal", "category": "CargoType"}, {"entity": "Rotterdam Port", "category": "Port"}, {"entity": "Holland Line", "category": "Company"}, {"entity": "10 October 2023", "category": "Date"}]

reducing the manual intervention and pattern design typically required for traditional knowledge graph construction. Meanwhile, with the Knowledge Alignment Mechanism, we can accurately align the knowledge extracted through OIE with domain-specific knowledge. By aligning at the conceptual knowledge level, the integration of open information with domain knowledge not only ensures semantic accuracy but also significantly enhances the efficiency of information extraction and the adaptability of the knowledge graph. Mathematically, we define the goal of conceptual knowledge alignment as minimizing the discrepancy between open information and domain knowledge:

$$\min \sum_{i=1}^{n} \mathrm{Dist}\left(O_i, K_i\right) \tag{10}$$

Here, O_i represents the ii-th piece of knowledge extracted through Open Information Extraction (OIE), K_i denotes the corresponding domain-specific knowledge, and $\mathrm{Dist}\left(O_i, K_i\right)$ represents the semantic discrepancy between them.

This study explores prompt design for the construction of a maritime domain knowledge graph. The prompts consist of several parts, including task descriptions, examples, and input text. The task description contains role definitions, entity definitions, and task settings, all aimed at maximizing the model's understanding of the task.

In the subsequent phase of the experiment, we extended our extraction framework to the relation extraction task. This phase aimed to identify semantic associations between the extracted entities, such as the route relationship between ships and ports, the transportation relationship between cargo and ships, and the ownership relationship between companies and ships. For this, we employed a schema-constrained, prompt-engineering-based approach to guide the large language model in understanding contextual relationships between entities for accurate relation identification.

To enhance the accuracy of relation extraction, we designed multi-level relation extraction prompt templates, tailored to specific maritime domain relationships. For example, prompts for "Ship-Port" relationships included contextual cues to extract critical information such as departure and destination ports, while "Company-Ship" relationship prompts focused on identifying ownership and operational rights. We further employed a few-shot learning strategy, providing typical examples of relation extraction tasks to improve the model's understanding of domain-specific relationship patterns.

3.7. Reason and Query

The following pseudocode illustrates the reasoning process in knowledge graph-enhanced retrieval when querying with natural language:

This algorithm employs a structured methodology to address a main question by breaking it down into sub-questions, processing them iteratively, and aggregating their answers to generate a comprehensive response. The algorithm's workflow can be described as follows:

Algorithm 2: Question Answering Workflow with Sub-Question Derivation

Input : Main question Q, Predefined schema *Schema*
Output: Answer to main question
// Step 1: Extract SPO structures and match related documents
$SPO\_structures \leftarrow \text{Extract\_SPO}(Q)$
$documents \leftarrow \text{Match\_Documents}(SPO\_structures)$
// Step 2: Derive sub-questions from the main question
$sub\_questions \leftarrow \text{Derive\_Sub\_Questions}(Q)$
// Step 3: Process each sub-question
$memory \leftarrow \{\}$; // Initialize memory to store previous answers
for $idx = 0$ **to** $len(sub\_questions) - 1$ **do**
  $sub\_question \leftarrow sub\_questions[idx]$
  if $idx == 0$ **then**
    // First sub-question processing
    $entities \leftarrow \text{Extract\_Entities}(sub\_question)$
    $schema\_match \leftarrow \text{Match\_Schema}(entities, Schema)$
    $recall\_text \leftarrow \text{Recall\_Text}(entities, schema\_match)$
    $answer \leftarrow \text{Infer\_Answer}(sub\_question, recall\_text)$
    $memory[idx] \leftarrow answer$; // Store the answer in memory
  end
  else
    // Subsequent sub-question processing using memory and recalled documents
    $documents\_from\_memory \leftarrow \text{Recall\_Documents\_From\_Memory}(memory)$
    $answer \leftarrow$ $\text{Infer\_Answer\_From\_Documents}(sub\_question, documents\_from\_memory)$
    $entities \leftarrow \text{Extract\_Entities}(answer)$
    $schema\_match \leftarrow \text{Match\_Schema}(entities, Schema)$
    $new\_text \leftarrow \text{Recall\_Text}(entities, schema\_match)$
    $new\_answer \leftarrow \text{Infer\_Answer}(sub\_question, new\_text)$
    $memory[idx] \leftarrow new\_answer$; // Store the new answer in memory
  end
end
// Step 4: Aggregate answers from all sub-questions
$final\_answer \leftarrow \text{Aggregate\_Answers}(memory)$
return $final\_answer$

1) Input and Initialization

The algorithm takes as input the main question Q and a predefined schema *Schema*, which provides structural guidelines for knowledge representation and reasoning. The output is the inferred answer to the main question.

2) SPO Extraction and Document Matching

The algorithm begins by extracting subject-predicate-object (SPO) structures from the main question using the function *Extract_SPO*. These SPO structures serve as key semantic units to identify and retrieve related documents through *Match_Documents*, thereby establishing a document corpus relevant to the question.

3) Sub-Question Derivation

The main question is decomposed into some sub-questions using *Derive_Sub_Questions*. These sub-questions allow for a granular exploration of the problem, facilitating a modular approach to information retrieval and inference.

4) Iterative Sub-Question Processing

First Sub-Question: Entities are extracted using *Extract_Entities* and matched

against the schema with *Match_Schema*. The matched schema guides the retrieval of relevant text through *Recall_Text*, which is subsequently used to infer an answer via *Infer_Answer*. The inferred answer is stored in memory for subsequent use.

Subsequent Sub-Questions: For each sub-question, documents and answers from prior iterations are recalled using *Recall_Documents_From_Memory*. Answers are inferred iteratively, leveraging newly recalled entities and schema matches. Each new answer is stored in memory, enhancing the knowledge base incrementally.

1) Answer Aggregation

Once all sub-questions are processed, their respective answers are aggregated using *Aggregate_Answers*. This step synthesizes the information into a unified final answer to the main question.

2) Output

The final answer, representing an integrated resolution to the main question, is returned as the output.

The proposed algorithm exhibits several innovative features, including schema-based reasoning, which ensures domain alignment and enhances semantic coherence across sub-questions, and an iterative memory mechanism that stores and recalls intermediate answers to improve contextual understanding and reasoning over multiple sub-questions. Additionally, the dynamic text recall process, guided by schema matching and iterative entity extraction, ensures the retrieval of relevant and precise information for subsequent inferences. The modular design of the workflow, facilitated by sub-question derivation, enables parallelization and targeted processing, thereby enhancing adaptability for addressing complex queries in structured domains. These features collectively contribute to the algorithm's robustness and effectiveness in handling hierarchical question-answering tasks.

4. Experiment

4.1. Dataset

The dataset integrates four primary types of data, each carefully curated to support comprehensive coverage of the maritime industry and its regulations, operations, and research needs.

The first category, Maritime Regulations and International Conventions, encompasses authoritative documents such as the International Convention for the Safety of Life at Sea (SOLAS) and the International Convention for the Prevention of Pollution from Ships (MARPOL). This dataset includes international regulatory texts and national shipping laws, focusing on areas such as maritime safety, environmental protection, and vessel management. With a volume of approximately 1.5 GB, the data predominantly exists in PDF and XML formats, ensuring structured and detailed representation of legislative requirements.

The second category, Ship Management Documents, derives from operational manuals, maintenance records, and voyage reports provided by crew companies and shipping enterprises. Comprising roughly 2 GB, these documents primarily use DOC and XLS formats, with some scanned images stored as PNG or JPEG files. This dataset offers rich insights into shipboard procedures and operational standards, facilitating practical applications in ship management and analytics.

The third type, Cargo and Port Data, includes cargo manifests, port operation workflows, and terminal facility information. This data originates from publicly available datasets provided by port operators and internal business data from enterprises. It totals approximately 1.2 GB and is formatted in CSV and JSON, enabling seamless integration and analysis of port-related logistics and infrastructure.

Lastly, Industry Research Reports provide strategic insights into market trends and technological developments. Sourced from maritime associations, academic journals, and industry white papers, this dataset covers dynamic aspects of the shipping industry. Comprising around 800 MB of data, it is stored mainly in PDF and HTML formats, offering an invaluable reference for understanding market and technology trends.

In the data preprocessing phase, rigorous cleaning was performed to remove duplicates and noise, ensuring consistency and relevance across all datasets. Document encoding was standardized to UTF-8, and semantic normalization was implemented—for instance, classifying "SOLAS Article 33" as a regulation clause and "Shanghai Port" as a port entity. Standardization of maritime terminology and uniformity in date-time formats further enhanced the data's usability.

After preprocessing, the consolidated dataset totals 5.5 GB, encompassing approximately 500,000 entities, 2 million relationships, and 300 million tokens of text corpus. This dataset is characterized by its diversity and high quality, offering comprehensive coverage of core maritime data sources. The systematic preprocessing steps ensure reliability and consistency, forming a robust foundation for constructing an efficient maritime knowledge graph.

4.2. Experiment Platform

The experimental platform is equipped with advanced hardware for high-performance computing. It features four Kunpeng 920 processors, each with 48 cores running at a clock speed of 2.6 GHz, providing a total of 192 cores. For acceleration, the system includes eight Ascend 910B AI cards, each equipped with 64 GB of HBM memory. The memory configuration includes 24 modules of 64 GB DDR4 memory, operating at a frequency of 3200 MHz, with a total capacity of 1.5 TB.

4.3. Experiment Results

To validate the effectiveness of this method, we conducted a series of comparative experiments designed to systematically assess the performance of various extraction approaches. The experimental datasets were sourced from a diverse range of

maritime domain regulatory texts, shipping reports, and tariff guidelines, covering both structured and unstructured data formats. Manually annotated data served as the Gold Standard, and performance was evaluated using standard metrics including Precision, Recall, F1-score, Coverage, and runtime.

The extraction methods compared in the experiments included the Schema-constrained and OIE-combined method (Schema + OIE), the Schema-based extraction method (Schema-only), and the OIE-based extraction method (OIE-only). The experimental results on the baseline dataset are presented in Table 2.

Table 2. Comparative experiment results on the baseline dataset.

METHOD	PRECISION	RECALL	F1 SCORE	Coverage	Average Runtime
Schema +OIE	0.93	0.95	0.94	90%	14.3
Schema-only	0.79	0.69	0.74	75%	21.8
OIE-only	0.70	0.62	0.66	82%	10.2

The experimental results on the baseline dataset (about 20 MB) demonstrate that the Schema + OIE method outperforms the others across multiple key metrics. This method achieved an F1-score of 0.94, a high recall rate of 0.95, and a coverage rate of 90%, while maintaining a reasonable processing efficiency with an average runtime of 14.3 seconds. In comparison, the Schema-only method exhibited a slight advantage in precision (0.95) but struggled with recall, achieving a significantly lower recall rate of 0.69 and a corresponding F1-score of 0.74. This indicates that the strict schema-based approach, while precise, leads to an overly narrow scope of extraction, resulting in a lower coverage of the data. The OIE-only method, while excelling in recall (0.95) and coverage (90%), suffered from lower precision (0.70) and an F1-score of 0.66, as it lacked domain-specific constraints, which resulted in the extraction of noisy and irrelevant information.

These findings highlight that the Schema + OIE approach effectively combines the domain-specific constraints of the schema with the flexibility of OIE, thus enhancing the accuracy and efficiency of knowledge extraction. This method is particularly suitable for the maritime domain, where both structured and unstructured data are prevalent, and domain-specific constraints are essential for accurate extractions. Additionally, the Schema + OIE approach minimizes reliance on manual annotation by leveraging prompt generation and few-shot learning strategies, which help in extracting information from diverse textual sources.

To further validate the robustness of this method, we extended the experiments to a larger and more diverse dataset containing maritime-related texts from over 10,000 documents, including international maritime regulations, vessel specifications, and cargo handling procedures. The result is presented in Table 3.

On this larger dataset, the Schema + OIE method maintained consistent performance with an F1-score of 0.92, a recall rate of 0.92, and a coverage of 92%. Notably, the runtime was optimized to 12.3 seconds, demonstrating the scalability

and efficiency of this approach when applied to larger datasets. The Schema-only method, while showing precision advantages (0.85), exhibited a decrease in recall (0.75) and a corresponding drop in coverage (73%) on the expanded dataset. This further underscores the limitations of a purely schema-based approach, particularly when applied to large-scale, diverse textual data, where rigid extraction rules cannot capture the full spectrum of knowledge. Meanwhile, the OIE-only method showed notable improvements in recall (0.93) and coverage (85%) but at the cost of precision (0.73), leading to a reduced F1-score of 0.67, indicating the need for more structured domain knowledge integration.

These experimental results strongly suggest that the Schema + OIE method not only achieves superior extraction performance but also demonstrates significant scalability on large datasets. This method's ability to effectively combine structured schema-based knowledge with flexible, open information extraction makes it highly suitable for real-world applications, especially in complex domains like maritime law and regulations.

Table 3. Performance of extraction methods on a larger maritime dataset (10,000+ documents).

METHOD	PRECISION	RECALL	F1 SCORE	Coverage	Average Runtime
Schema + OIE	0.93	0.92	0.92	92%	12.3
Schema-only	0.85	0.75	0.80	73%	20.5
OIE-only	0.73	0.63	0.67	85%	11.1

In summary, the experimental results demonstrate the significant advantages of the Schema + OIE method in both entity and relation extraction tasks, especially when applied to large-scale maritime datasets. The method's high accuracy, scalability, and efficiency make it an ideal solution for constructing knowledge graphs in the maritime domain, facilitating better knowledge discovery and decision-making.

5. Conclusion and Future Directions

This study proposes innovative solutions for optimizing knowledge graph retrieval in the shipping domain, particularly in the improvement of Retrieval-Augmented Generation (RAG) technology and knowledge graph construction. Despite significant advancements in modern Natural Language Processing (NLP) techniques, traditional retrieval methods still face numerous challenges in specialized fields, such as shipping, which are characterized by their strong professionalism and complex knowledge structures. The core contribution of this research lies in overcoming the limitations of traditional methods by integrating an adaptive schema generation mechanism with reinforcement learning models, thereby achieving precise and flexible knowledge representation and advancing the application of large language models in domain-specific knowledge retrieval.

Firstly, the graph-based inverted index method proposed in this paper plays a

crucial role in enhancing the efficiency and accuracy of information retrieval. By leveraging graph models, the semantic associations between documents are strengthened, leading to more accurate document ranking and relevance measurement. This improvement significantly enhances the model's responsiveness in specific domains. In the shipping industry, where the knowledge base is highly complex, traditional keyword-matching retrieval mechanisms often fail to capture the underlying semantic information of user queries. In contrast, the graph-based inverted index can effectively utilize the graph structure for deep semantic matching and reasoning, thus improving the accuracy of retrieval results.

Secondly, the introduction of the semi-structured method and the adaptive schema generation mechanism provides greater flexibility in constructing knowledge graphs. The top-down and bottom-up combined schema design strategy not only ensures the precise abstraction of core concepts in the shipping domain but also allows for dynamic adjustments to the knowledge graph structure based on actual application needs. By modeling the knowledge graph construction process as a Markov Decision Process, the study effectively automates the updating and optimization of knowledge representation, enabling the knowledge graph to continuously adapt to new information as the domain knowledge evolves. This dynamic update capability lays a solid foundation for the sustainable development of vertical domain knowledge graphs.

In terms of further technical integration, the study combines large language models (LLMs) with traditional information extraction techniques, employing self-attention mechanisms and Graph Attention Networks (GATs) to enhance the model's semantic understanding capabilities. Traditional information extraction methods often rely on rules and templates, which have limitations, especially in handling polysemous words and ambiguous queries. In contrast, self-attention mechanisms and GATs can capture semantic relationships globally and integrate information across multiple layers, significantly improving the model's reasoning and accuracy. This innovation not only enables the model to perform excellently in complex query scenarios within the shipping domain but also provides valuable insights for knowledge graph retrieval in other vertical domains.

Additionally, the logic-driven solver and symbolic decomposition mechanism introduced in this paper further enhance the model's ability to understand and reason about complex semantic relationships. Traditional language model-based reasoning often struggles with long inference chains and weak context associations. However, the symbolic decomposition technique used in this study strengthens logical derivation and context integration during the reasoning process, enabling the model to provide more accurate and reliable inference results when faced with complex problems. This is particularly important in the shipping domain, which involves extensive legal regulations and operational rules, where symbolic reasoning supports the integration and retrieval of such knowledge.

Despite the significant achievements in optimizing knowledge graph retrieval mechanisms, several areas warrant further exploration and improvement. Firstly,

the construction of knowledge graphs in the shipping domain remains highly specialized and complex, leaving room for enhancement in the modeling of domain-specific knowledge and semantic relationships. Future work could consider integrating domain expert knowledge with model training to deepen the model's understanding of domain-specific knowledge. Secondly, as knowledge graph and LLM technologies continue to advance, achieving more efficient cross-domain knowledge transfer and adaptive updates will be a key focus for future research. Ultimately, with the diversification of application scenarios, optimizing the real-time performance and scalability of the inference process will become an important direction for future research.

Furthermore, while knowledge graphs have demonstrated significant advantages in text understanding and information retrieval, their application in method function inference and invocation, which is more closely aligned with real-world shipping operations, presents several challenges. Method functions often involve abstract computational steps and complex logical reasoning, which are difficult to represent directly using traditional knowledge graph structures. The abstractness and diversity of functions limit the graph's ability to capture internal details and execution processes. Additionally, method function invocations typically depend on specific contextual environments, which are challenging to model accurately in static knowledge graphs. Function invocation involves not only input-output mapping but may also be influenced by data flow, control flow, and external factors, increasing the complexity of dynamic reasoning. On the other hand, knowledge graphs face challenges in handling multi-step reasoning and composite tasks. Complex tasks often require sequential calls or dependency inferences involving multiple functions, and these reasoning chains may span multiple domains, which traditional graph models struggle to integrate effectively. Moreover, the reasoning capabilities of knowledge graphs are constrained by the scale and structural complexity of the graph, particularly when dealing with implicit knowledge and cross-domain reasoning, where the graph may not provide sufficient contextual information.

To address these challenges, future research could explore the integration of knowledge graphs with more advanced technologies. For example, introducing Program Synthesis and Graph Neural Networks (GNNs) can better capture the invocation relationships and dynamic dependencies between method functions. Dynamic knowledge graph construction can help solve context dependency issues by updating and adjusting the graph structure in real-time to adapt to the changing environmental factors in function invocations. Furthermore, combining Neural-Symbolic Reasoning and Reinforcement Learning methods can further enhance the system's capabilities in handling complex multi-step reasoning and adapting to method function invocations.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Chen, J., Zhang, X., Xu, L. and Xu, J. (2024) Trends of Digitalization, Intelligence and Greening of Global Shipping Industry Based on CiteSpace Knowledge Graph. *Ocean & Coastal Management*, **255**, Article ID: 107206. https://doi.org/10.1016/j.ocecoaman.2024.107206

[2] Wan, H., Fu, S., Zhang, M. and Xiao, Y. (2023) A Semantic Network Method for the Identification of Ship's Illegal Behaviors Using Knowledge Graphs: A Case Study on Fake Ship License Plates. *Journal of Marine Science and Engineering*, **11**, Article 1906. https://doi.org/10.3390/jmse11101906

[3] Zhang, Y., Xu, R., Lu, W., Mayer, W., Ning, D., Duan, Y., *et al.* (2023) Multi-Modal Spatio-Temporal Knowledge Graph of Ship Management. *Applied Sciences*, **13**, Article 9393. https://doi.org/10.3390/app13169393

[4] Li, Y., Liu, X., Wang, Z., Mei, Q., Xie, W., Yang, Y., *et al.* (2024) Construction of a Large-Scale Maritime Element Semantic Schema Based on Knowledge Graph Models for Unmanned Automated Decision-making. *Frontiers in Marine Science*, **11**, Article 1390931. https://doi.org/10.3389/fmars.2024.1390931

[5] Zhang, Z.Y., *et al.* (2019) ERNIE: Enhanced Language Representation with Informative Entities. arXiv: 1905.07129.

[6] Liu, W., Zhou, P., Zhao, Z., Wang, Z., Ju, Q., Deng, H., *et al.* (2020) K-BERT: Enabling Language Representation with Knowledge Graph. *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**, 2901-2908. https://doi.org/10.1609/aaai.v34i03.5681

[7] Sun, T., Shao, Y., Qiu, X., Guo, Q., Hu, Y., Huang, X., *et al.* (2020) CoLAKE: Contextualized Language and Knowledge Embedding. *Proceedings of the* 28*th International Conference on Computational Linguistics*, Barcelona, December 2020, 3660-3670. https://doi.org/10.18653/v1/2020.coling-main.327

[8] Xiong, W.H., Du, J.F., Wang, W.Y. and Stoyanov, V. (2019) Pretrained Encyclopedia: Weakly Supervised Knowledge-Pretrained Language Model. arXiv: 1912.09637.

[9] Yao, L., Mao, C.S. and Luo, Y. (2019) KG-BERT: BERT for Knowledge Graph Completion. arXiv: 1909.03193.

[10] Yu, D., Zhu, C., Yang, Y. and Zeng, M. (2022) JAKET: Joint Pre-Training of Knowledge Graph and Language Understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, **36**, 11630-11638. https://doi.org/10.1609/aaai.v36i10.21417

[11] Wang, R., Tang, D., Duan, N., Wei, Z., Huang, X., Ji, J., *et al.* (2021) K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters. *Findings of the Association for Computational Linguistics*: *ACL-IJCNLP* 2021, August 2021, 1405-1418. https://doi.org/10.18653/v1/2021.findings-acl.121

[12] Zhang, H.Y., *et al.* (2023) Research on Question Answering System on Joint of Knowledge Graph and Large Language Models. *Journal of Frontiers of Computer Science & Technology*, **17**, 2377. https://doi.org/10.3778/j.issn.1673-9418.2308070

[13] Peters, M.E., *et al.* (2019) Knowledge Enhanced Contextual Word Representations. arXiv: 1909.04164.

[14] He, B., Zhou, D., Xiao, J., Jiang, X., Liu, Q., Yuan, N.J., *et al.* (2020) BERT-MK: Integrating Graph Contextualized Knowledge into Pre-Trained Language Models. *Findings of the Association for Computational Linguistics*: *EMNLP* 2020, November 2020, 2281-2290. https://doi.org/10.18653/v1/2020.findings-emnlp.207

[15] Yang, A., Wang, Q., Liu, J., Liu, K., Lyu, Y., Wu, H., *et al.* (2019) Enhancing Pre-Trained Language Representations with Rich Knowledge for Machine Reading

Comprehension. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, July 2019, 2346-2357. https://doi.org/10.18653/v1/p19-1226

[16] Huang, Z.H., *et al.* (2015) Bidirectional LSTM-CRF Models for Sequence Tagging. arXiv: 1508.01991.

[17] Vaswani, A., *et al.* (2017) Attention is All you Need. arXiv: 1706.03762.

[18] Mao, A.Q., Mohri, M. and Zhong, Y.T. (2023) Cross-Entropy Loss Functions: Theoretical Analysis and Applications. arXiv: 2304.07288.

[19] Degris, T., White, M. and Sutton, R.S. (2012) Linear Off-Policy Actor-Critic. arXiv: 1205.4839.

[20] Liang, L., Sun, M., Gui, Z., *et al.* (2024) KAG: Boosting LLMs in Professional Domains via Knowledge Augmented Generation. arXiv: 2409.13731.

Appendix A.

Table A1. The schema of underlying concepts of maritime transportation.

Entity	Properties	Description
Chunk	Content: Text Description: Text	A chunk refers to a segment or piece of information. It can be a group of words that form a meaningful unit, such as a noun phrase or a verb phrase, or it can refer to a block of data in computing contexts.
Artificial object	Description: Text Semantic Type: Text	An artificial object is any object that is created or produced by humans, as opposed to natural objects, which occur without human intervention. Artificial objects can range from simple tools and everyday items to complex machines and structures.
Building	Description: Text Semantic Type: Text	A building is a structure with walls and a roof, typically designed for human occupancy or use. Buildings can serve a wide range of purposes, including residential, commercial, industrial, and institutional.
Creature	Description: Text Semantic Type: Text	A creature is a living being, often used to refer to animals, but it can also include other forms of life. The term "creature" is broad and can encompass a wide range of organisms, from microscopic life forms to large mammals.
Concept	Description: Text Semantic Type: Text	A concept is a general idea, notion, or abstraction that represents a class of objects, attributes, actions, or relationships. Concepts are fundamental to human cognition and communication, allowing us to categorize and understand the world around us. They are often expressed through words, symbols, or other forms of representation.
Date	Description: Text Semantic Type: Text	A date refers to a specific day in a calendar, typically expressed in terms of the day, month, and year. Dates are used to mark and organize events, appointments, and historical records. They are essential for timekeeping, scheduling, and documenting when something occurs.
Geographic Location	Description: Text Semantic Type: Text	A geographic location refers to a specific place or position. It can be described in various ways, depending on the context and the level of detail required.
Organization	Description: Text Semantic Type: Text	An organization is a structured group of people who come together to achieve a common goal or set of goals. Organizations can be formal or informal and can vary widely in size, structure, and purpose.
Person	Description: Text Semantic Type: Text	A person is an individual human being. The concept of a person encompasses various aspects, including biological, psychological, and social dimensions.
Transport	Description: Text Semantic Type: Text	Transport, or transportation, refers to the movement of people, animals, and goods from one place to another. It is a critical component of modern society, enabling trade, travel, and communication.
Event	Description: Text Semantic Type: Text	An event is a specific occurrence or happening, often marked by a particular time and place. Events can be planned or spontaneous and can range from personal milestones to large-scale public gatherings.
Others	Description: Text Semantic Type: Text	Others denote categories beyond these top-level classes, and it leaves room for expansion for entities that are difficult to categorize.

Table A2. The schema of specialized concepts of maritime transportation.

Entity	Properties	Relations	Constraints
Port	Description: Text	Connected To: Port (represents transit ports or connections)	Port Name Constraints: Enum = $Names of all ports in the world
	Port Name: Text	Located In: Geographic Region	Unique Constraint: Port Name must be unique
	Location: Geographic Location		
	Capacity: Float		Capacity ≥ 0
	Facilities: Text		
Company	Description: Text	Operates: Vessel, Shipping Route	Unique Constraint: Company Name must be unique
	Company Name: Text		
	Address: Geographic Location		
Vessel	Description: Text	Operated By: Company, Visits: Port	Vessel Name Constraints: Enum = $Names of all vessels
	Vessel Name: Text		Vessel Name must be unique
	Vessel Type: Text		Vessel Type Constraints: Enum = "Container, Tanker, Bulk Carrier, Others"
	Vessel Capacity: Float		Capacity ≥ 0
Shipping Route	Description: Text	Operated By: Company, Includes: Port	Duration Constraint: Must be > 0
	Route Name: Text		Route Name must be unique
	Origin Port: Port		
	Destination Port: Port		
	Distance: Float		Distance > 0
	Duration: Text		
	Capacity: Float		Capacity ≥ 0
	Frequency: Text		
Supplier	Description: Text	Supplies To: Customer	
	Supplier Company: Company		
	Contact Person: Person		
Customer	Description: Text	Purchases From: Supplier	
	Customer Company: Company		
	Contact Person: Person		
Risk	Description: Text	Affects: Port, Vessel, Company	Risk Type Constraints: Enum = "Financial, Operational, Reputational, Compliance"
	Risk Type: Text		
	Mitigation Plan: Text		

Continued

Task	Description: Text	Assigned To: Person, Related To: Port, Vessel	Task Priority Constraint: Enum = "High, Medium, Low"
	Task Name: Text		Task Name must be unique
	Task Description: Text		
Document	Description: Text	Related To: Port, Vessel, Company, Task	Document Expiry Date Constraint: Must be after Issue Date
	Document Title: Text		Document Title must be unique
	Document Content: Text		
	Document Date: Date		
Cargo Type	Description: Text	Transported By: Vessel	Cargo Type Constraints: Enum = "Container, Liquid, Hazardous, General"
	Category: Text		