Scientific Reports

https://doi.org/10.1038/s41598-026-38066-w

Article in Press

The construction and refined extraction techniques of knowledge graph based on large language models

Received: 29 May 2025

Accepted: 28 January 2026

Published online: 10 February 2026

Cite this article as: Peng L., Yang P., Juexiang Y. *et al.* The construction and refined extraction techniques of knowledge graph based on large language models. *Sci Rep* (2026). https://doi.org/10.1038/s41598-026-38066-w

Li Peng, Pei Yang, Ye Juexiang & Li Yuangan

The Construction and Refined Extraction Techniques of Knowledge Graph Based on Large Language Models

Li Peng[a,b]*, Pei Yang[a], Ye Juexiang[c], Li Yuangan[d]

[a]*Northwestern Polytechnical University, Xi'an, China*
[b]*Chinese Aeronautical Establishment, Beijing, China*
[c]*Harbin Institute of Technology, Harbin, China*
[d]*Beihang University, Beijing, China*

Abstract

With the growing need for intelligent decision-support systems, the development of high-quality knowledge graphs has become essential for improving operational efficiency and decision reliability. However, the specialized nature, distributed sources, and sensitive aspects of this knowledge present unique challenges to conventional knowledge management approaches. Current general-purpose large language models often struggle with domain-specific text comprehension, particularly in accurately interpreting technical parameters and operational guidelines. To address these limitations, this paper introduces a framework for building and refining specialized knowledge graphs using adapted large language models. Our approach involves fine-tuning base LLMs with domain-specific datasets, enabling them to better handle complex terminology and semantic nuances. The framework incorporates a multimodal knowledge integration pipeline that combines rule-based systems with ontological structures to extract and link entities from diverse data sources, creating an adaptive knowledge network. Experimental results demonstrate that our fine-tuned model achieves substantial gains in relationship extraction accuracy, while the resulting knowledge graph shows strong performance in semantic coherence and operational reasoning assessments, offering robust support for critical decision-making processes. This research presents a novel approach for effective knowledge integration and cross-functional collaboration in specialized domains.

Keywords: Knowledge Graph, Adapted Large Language Model, Multimodal Knowledge Integration, Operational Decision Support, Dynamic Knowledge Network

1. Introduction

In modern operational decision-making systems, the demand for intelligent support has surged due to increasingly complex environments and accelerated technological evolution. The effectiveness of critical decision processes directly impacts strategic outcomes, particularly in scenarios requiring multi-domain coordination where commanders must rapidly synthesize information from vast, heterogeneous datasets to formulate precise operational plans. Domain knowledge is typically dispersed across operational manuals, technical documentation, sensor data, and historical case studies, encompassing specialized content including system parameters, operational guidelines, and environmental intelligence. This fragmentation creates integration challenges that can compromise decision quality.

* Corresponding author: Li Peng. Tel.: +86-18210260981.
E-mail address: fubin@buaa.edu.cn.

Knowledge graphs (KGs) offer a structured solution [1], representing information as interconnected entity-relationship triples (head, relation, tail) that form semantic networks. While successful in general domains (WordNet, DBpedia [2-5]) and applications like information retrieval [6-8], KG construction in specialized contexts presents unique challenges: (1) highly distributed and dynamic knowledge; (2) limited data accessibility; (3) real-time update requirements; and (4) domain-specific semantic complexity.

However, Constructing knowledge graphs in specialized domains presents notable challenges: the knowledge is highly dispersed and dynamic, encompassing heterogeneous data such as technical parameters, operational rules, and spatial intelligence; data sensitivity limits public resource availability, with critical information often residing in restricted internal documents; operational logic and system status can change in real time according to field conditions, making traditional static knowledge graphs unsuitable for dynamic updates; and the domain-specific terminology causes general-purpose NLP models to have limitations in semantic understanding.

Existing methods exhibit significant shortcomings in specialized knowledge management. While general-purpose LLMs possess strong text comprehension capabilities, their accuracy in entity and relationship extraction drops significantly when dealing with domain-specific terms (e.g., technical codes, operational abbreviations) and unstructured text (e.g., spatiotemporal descriptions in reports) [9]. Traditional rule engines rely on manually defined patterns, making adaptation to dynamic data difficult [10], while statistical learning-based models are constrained by scarce labeled data and limited domain transferability [11]. Additionally, data sensitivity complicates cross-departmental knowledge sharing, exacerbating information silos. In this context, constructing a high-quality, dynamically updated knowledge graph and ensuring its reliable application in decision-making becomes a core challenge.

To address these issues, this paper proposes a knowledge graph construction framework that integrates domain-adapted LLMs with multimodal knowledge fusion. First, a general-purpose LLM is fine-tuned using domain-specific corpora to enhance its ability to identify entities and complex relationships, such as technical specifications, operational rules, and environmental factors. For example, the model can parse implicit compatibility logic in manuals or extract spatiotemporal relations from reports. Second, a multimodal knowledge extraction pipeline integrates text, images, structured databases, and other multi-source data, combining ontology constraints and rule engines to build a dynamic knowledge network. Finally, the knowledge graph is validated using real-world exercise data, with evaluation experiments assessing semantic consistency, reasoning support, and update efficiency. While the architecture defines adapters for images, tables/structured sources, and time-series logs, the current experiments activate the text branch only.

The main contributions of this study are as follows:

(1) Task-aware domain fine-tuning protocol. We introduce a Knowledge Routing Network that guides a hierarchical LoRA schedule for domain tasks, selecting adapters and parameter scopes to turn parameter-efficient tuning into a task-aware adaptation mechanism—beyond the independent use of LoRA, CoT, or RAG.

(2) Privacy-preserving dataset generation pipeline. We design a desensitization workflow for real exercise data—covering entity generalization, functional coding, and controlled masking—to produce training and evaluation sets that preserve utility while meeting security constraints.

(3) Transparent graph-quality assessment and attribution. We report precision/recall/F1 for entity and relation extraction and coverage/structure metrics of the graph (average degree, edge density, clustering coefficient, update latency), together with a reproducible ablation protocol that attributes gains to individual modules rather than to generic techniques.

The rest of this paper is organized as follows: Section 2 reviews the current research on domain knowledge graphs and large language models; Section 3 details the domain

fine-tuning methods and knowledge graph construction framework; Section 4 presents the implementation of the graph and results from multidimensional evaluation experiments based on real data; Section 5 summarizes the research findings and discusses future directions.

2. Related Work

2.1. Classical Knowledge Graph Construction Methods

Knowledge graph construction has evolved from manual efforts to intelligent automation. Early methods relied on expert knowledge; for example, WordNet [12][13] defined semantic relationships through linguistic annotation, yielding high accuracy but low scalability. In structured or semi-structured scenarios, rule-based methods like DBpedia [14][15] extracted triples from Wikipedia infoboxes using predefined rules. Though efficient for fixed-format data, such approaches struggle with the complex semantics of natural language [35].

With deep learning advancements, neural network-based techniques have enhanced automation [16][17]. In NER tasks, tools such as SpaCy, NLTK, and ltp [18][19] blend rules and statistical models, while BiLSTM-CRF [20] improves sequence labeling via contextual learning. Pre-trained models like BERT [21] further optimize performance, especially in cross-lingual settings [22][23], and domain-specific fine-tuning [24–27] significantly enhances the recognition of specialized terminology.

For RE tasks, CNNs [28][29] improve classification by extracting local features, while distant supervision [30] enables automatic labeling but introduces noise. Sentence-level attention mechanisms [31] reduce this by weighing relevant context. Recent frameworks incorporating entity masking and contrastive pretraining [32] further enhance robustness. PEFT-based methods [33] and tools like OpenNRE [34] offer scalable and adaptable RE solutions.

Despite progress, limitations remain in domain applications: manual and rule-based methods cannot handle large-scale, unstructured data or deep semantics [35]; deep models require labeled data, which is scarce in restricted domains; and traditional full-parameter tuning is costly and lacks efficiency [36]. This paper proposes integrating LLMs to overcome these barriers in specialized KG construction.

2.2. Domain Adaptation of LLMs

Large-scale pre-trained LLMs have become a core paradigm in NLP due to their powerful semantic capabilities. Models like GPT-4 and LLaMA-3, with trillions of parameters, learn cross-task generalization via large-scale pretraining and subsequent task-specific fine-tuning [37]. However, as model size grows, full-parameter fine-tuning faces issues such as high GPU memory demand and significant computational overhead. For instance, fine-tuning LLaMA2-7B requires around 60GB of GPU memory [38], and storage burdens increase sharply in multi-task contexts. Moreover, full updates can degrade the model's general knowledge [39], limiting effectiveness in specialized domains. To overcome these challenges, Parameter-Efficient Fine-Tuning (PEFT) has gained prominence by updating only a small fraction of parameters.

LoRA inserts low-rank matrices into Transformer layers, enabling adaptation with just 0.1% of the total parameters. Hu et al. [40] applied LoRA to GPT-3 175B, reducing trainable parameters by 10,000x while retaining inference speed. This method has been deployed in scenarios like GPT4Tool for tool invocation [41]. Adapter modules embed lightweight networks into model layers, as in Houlsby et al.'s BERT extension [42], requiring under 1% extra parameters. However, stacking adapters may increase inference latency. Prefix-tuning, as introduced by Li et al. [43], optimizes continuous prefix vectors to guide outputs, achieving strong performance in low-resource tasks.

Other methods such as soft prompts [44], BitFit [45], and QLoRA [46] balance memory efficiency and task adaptability through selective parameter updates and quantization.

Beyond fine-tuning algorithms, domain adaptation also depends on effective knowledge injection and architecture design. In rail transportation, a high-speed rail maintenance KG improves fault diagnosis via multi-level completion [47]. For infrastructure management, BMKG applies graph mining to support classification and decision-making [48]. In safety analysis, knowledge graphs uncover causal accident chains using semantic reasoning [49]. Integrating domain triples as pseudo-text [50] or combining multimodal data like sensor logs [51] enhances model specialization and cross-modal understanding.

Recent studies further adapt large models for open-domain extraction. UniversalNER [55] introduces targeted distillation from LLMs to improve open NER under limited supervision, while BANER [56] leverages boundary-aware strategies to enhance few-shot entity recognition. For relations, Wang et. al. [57] explores cooperating LLMs with phrase-level probabilistic modeling for open relation extraction. These methods advance NER/RE capabilities at the instance level. Our work is complementary but differs in scope: we use a Knowledge Routing Network (KRN) to guide hierarchical LoRA for domain tasks and couple LLM extraction with ontology and rule constraints, triplet validation, and graph-level quality metrics and ablation attribution, targeting a reproducible KG construction–validation pipeline in sensitive domains.

Despite progress, sensitive domains still face barriers due to data access limitations. Tasks such as complex information extraction or situational analysis often involve unstructured and restricted data, limiting large-scale model training. Future research should focus on privacy-preserving fine-tuning, structured knowledge injection, and logic-constrained optimization to enable the secure and efficient deployment of LLMs in high-stakes application scenarios.

2.3. Knowledge Graph Construction Based on LLMs

In recent years, large-scale language models (LLMs) such as GPT-4, LLaMA, and PaLM have become key enablers of automated knowledge graph (KG) construction, owing to their strong semantic understanding and reasoning capabilities [37]. Traditional KG construction methods—based on manual annotation, rule engines, or small-scale pre-trained models—suffer from high cost and poor scalability. In contrast, LLMs leverage vast pretrained corpora to extract structured knowledge from unstructured text, offering a new paradigm. For instance, GPT-3 has shown near-expert performance in open-domain relation extraction, particularly in handling long-tail semantics, outperforming supervised models [38]. This has spurred research into prompt-based and in-context learning methods to guide LLMs in entity recognition, relation extraction, and logical validation, reducing reliance on labeled data.

LLM-based frameworks often reframe KG construction as a text-to-structure task. REBEL [52], for example, directly generates entity-relation triples without predefined ontologies or rules, achieving 1.8× the coverage of traditional approaches on Wikipedia. Another line of work integrates symbolic reasoning with LLMs through Chain-of-Thought (CoT) prompting, enabling interpretable, step-wise extraction of triples [53].

In specialized domains, LLM-driven KG construction has shown promising results. ClinicalKG [54] parses EHRs using GPT-4 to build a disease-symptom-drug network, achieving 89% accuracy in FDA-level drug interaction evaluations. FinGraph [55] extracts dynamic financial relationships and market risks from reports and news to support regulatory and investment decisions. In the industrial domain, a high-speed railway maintenance KG employs a multi-level KBGC framework and LLM-based log parsing to uncover equipment states and fault chains.

Despite these advances, applying LLMs in high-security or domain-constrained contexts remains challenging. General LLMs often underperform in specialized information extraction. In particular, KG construction in certain restricted domains is

still in an exploratory phase, lacking mature methodologies. This study aims to bridge that gap by integrating LLM language capabilities, domain adaptation techniques, and CoT reasoning with traditional KG methods to design an automated framework tailored to secure and specialized knowledge environments.

3. Methods

3.1. Parameter-efficient Fine-tuning of LLMs in the Domain

3.1.1. Construction of a Multi-source Corpus

This study focuses on the training requirements of large-scale pre-trained models in the domain, aiming to construct a high-quality, structured training corpus that can support multi-task adaptation. Successful deployment of LLMs in the domain relies on datasets that are rich in background information, highly reliable, and well-structured. These datasets must not only cover the complex needs of tasks such as tactical decision support, threat assessment, and related-knowledge question answering, but also establish a robust knowledge-sharing mechanism within the dataset. To this end, this study integrates multi-source, heterogeneous data into a unified training dataset, adopting a standardized data architecture to ensure that the training set can comprehensively support instruction fine-tuning, multi-task joint training, and continuous learning requirements.

In the data collection process, this study integrates various data sources to ensure that the corpus fully covers the critical tasks of tactical planning, equipment configuration, threat assessment, and communication command parsing. These data include tactical command communication logs, equipment technical documents, battlefield simulation data, and theoretical literature. The tactical command communication logs are transformed into instruction chains with time-domain labels through multi-level semantic parsing techniques, while incorporating tactical background descriptions and execution feedback records, providing complete information about the instruction execution process to assist in subsequent task analysis and optimization. The equipment technical documents are processed to construct matching rules between equipment performance and combat environments, providing key data support for equipment configuration decision-making tasks. The battlefield simulation data is decomposed into decision tree structures, with each node labeled with the probability of selection conditions and expected outcomes, effectively supporting the training requirements of tactical decision-making. The theoretical literature is structurally analyzed and converted into tactical rule explanation texts, establishing traceable logical mappings with historical campaign databases, providing a solid foundation of knowledge, especially crucial for question answering tasks.

While ensuring data diversity, this study implements a stringent desensitization system in the data processing phase, considering the high sensitivity of data. All raw data undergo semantic-level reconstruction and desensitization. Specifically, entities are standardized through a generalized transformation guided by a knowledge graph. For example, geographical coordinates are converted into relative position descriptions, unit organizational identifiers are transformed into functional operational unit codes, and equipment technical parameters are mapped to a standardized grading system. To further enhance data security and privacy protection, this study also integrates a virtual adversarial simulation engine, which reconstructs the abstract expression of core tactical logic through a probabilistic masking mechanism, ensuring that the original data distribution features are protected while reducing the risk of data leakage. The processed data undergoes semantic coherence verification and logical conflict detection to ensure data purity and quality, ultimately forming high-quality semantic units.

In terms of data architecture design, this study adopts an enhanced nested JSON structure to ensure the data can efficiently and flexibly support multi-task joint training.

Table1. Details of the dataset design

Task Category	Function Description	Data Proportion	Core Characteristics
Q&A	Answer questions about tactical principles and equipment characteristics	40%	Includes authoritative clause references and multi-condition applicable rule descriptions
Tactical Planning	Generating decision paths that comply with combat orders	25%	Includes stage goal breakdown, resource allocation plans, and contingency plans
Threat Assessment	Dynamically quantifying battlefield risk factors	20%	Integrated multi-source intelligence in a matrix evaluation model with a weighted scoring system
Equipment	Optimizing technical parameter	10%	Based on task objectives, equipment performance

Each data unit contains three main modules: the global situational framework, the task branch set, and the cross-domain indexing system. The global situational framework standardizes the description of the battlefield environment, with core fields including the classification of operational stages, dynamic enemy-force comparison, geographical constraint matrices, and task goal decomposition trees. These fields provide the model with necessary tactical context, help establish a shared semantic foundation across different tasks, and enhance multi-task adaptation capabilities. Specifically, the classification of operational stages helps the model understand the temporal features of tasks, the dynamic enemy-force comparison enables the model to quickly grasp the battlefield power dynamics, the geographical constraint matrices offer geographical limitations within the operational area, and the task goal decomposition trees break down the task into detailed steps, providing structured support for executing different tasks.

The task branch set divides different input-output structures according to the functional type of each task, focusing on supporting five core tasks: tactical planning, threat assessment, equipment configuration decision-making, instruction parsing, and question answering. The training data for each task is specifically designed according to its functional requirements. For example, the tactical planning task needs to generate decision paths that comply with operational orders, so the task branch includes details such as stage goal breakdown, resource scheduling plans, and contingency plans. The threat assessment task focuses on constructing a dynamic risk coefficient matrix, integrating multi-source intelligence and risk assessment models, providing the model with precise battlefield situation judgment. The instruction parsing task transforms natural language commands into executable operation codes, helping the model achieve automatic execution and optimization of tactical commands. The question answering task relies on high-quality knowledge sources, such as operational orders, equipment technical white papers, and historical campaign reviews, combining manual expert reviews and automated knowledge extraction to generate high-quality Q&A pairs, ensuring the standardization and credibility of the samples. Each task branch not only specifies the input-output format of the task but also provides in-depth descriptions of different functional needs, ensuring the accuracy and flexibility of the training model in various task scenarios.

The cross-domain indexing system establishes knowledge-sharing pathways between tasks through the global situational framework, allowing context to permeate the training processes of other tasks. This design not only significantly enhances task

coordination but also optimizes data flow and knowledge sharing during training, helping the model better understand the inherent relationships between different tasks, thus further improving its overall performance.

A typical training data sample intuitively demonstrates the corpus's ability to support multiple tasks. In this sample, the functionality of all modules is effectively reflected. The global situational module provides a complete description of battlefield environment features and force composition, while the task branch module defines input constraints and output specifications for different tasks through structured parameters, particularly in the design of question answering and tactical planning tasks. The task includes not only authoritative answers to specific tactical questions but also binds relevant legal references and dynamic adjustment rules; the tactical planning task refines key nodes of action steps through a standardized hierarchical structure. This sample shows how data structuring ensures that different task types can share basic battlefield situation data and independently define functional requirements for specialized domains, significantly reducing the complexity of multi-task training and

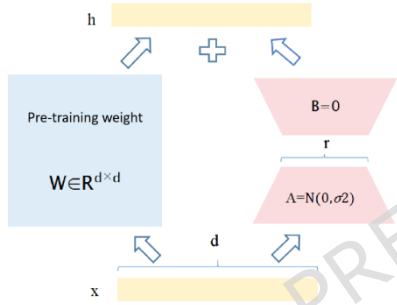

Fig1. The principle of LoRA

optimizing task coordination effects.

Table 1 further supplements the specific details of the dataset design, showing the proportion and core features of each task. The task proportions and core features in the table indicate the weight and requirements of each task in the overall dataset. For example, the question answering task accounts for 40% of the corpus, reflecting its dominant position in the entire training system. In this task, authoritative clause references and multi-condition applicable rule descriptions are its core features. The proportions of the tactical planning and threat assessment tasks are next, at 25% and 20%, respectively, highlighting the importance of tactical decision-making and situational assessment in applications. Equipment configuration decision-making and communication instruction parsing tasks have smaller proportions, at 10% and 5%, but they still hold significant value in specific application scenarios.

The setting of task proportions and core features ensures the balance and professionalism of the dataset in various task domains, while providing theoretical support for data allocation and task prioritization during training. Through refined task planning and data partitioning, this study not only ensures that different task types can share basic battlefield situation data but also allows them to independently define functional requirements in their respective fields. This significantly reduces the complexity of multi-task training and effectively improves the model's training efficiency and generalization ability. The combination of this design concept and the table content reflects the supportive relationships and differentiated distribution between tasks, thereby enhancing the training quality of each task module and the accuracy of task execution.

Through the approach outlined above, this study has constructed a unified training dataset capable of effectively supporting multi-task training for domain LLMs. This

dataset ensures in-depth integration of multi-source intelligence while systematizing knowledge through a structured data production process, providing a solid foundation for artificial intelligence applications in the domain. Particularly in the multi-task joint training model, the corpus design guarantees adaptability and data sharing, significantly enhancing the generalization ability and decision-making accuracy of the model. It is anticipated that this dataset will improve tactical decision-making capabilities and operational efficiency.in intelligent decision-support systems.

The datasets originate from sensitive exercises, so we apply a privacy-preserving workflow before any model access: entity generalization and pseudonymization, functional coding of organizational units, probabilistic masking or light rephrasing of high-risk snippets, and manual review. To balance collaboration and privacy, processing occurs in a controlled environment with role-based access, audit logging, and data minimization. External partners access de-identified corpora under data-use agreements; model artifacts, prompts, and code are shareable, while raw data remain on-premise. For cross-site validation, we use a “bring-the-code-to-the-data” or federated run approach so that only aggregate metrics leave the site. We monitor privacy using k-anonymity (≥ 5) and ℓ-diversity (≥ 2) checks for any released subsets, and apply aggregate-only reporting where required.

3.1.2. LoRA Adaptation Framework for Domain Knowledge

The hierarchical LoRA adaptation framework proposed in this study is based on the hierarchical nature of domain knowledge, with a structured low-rank parameter architecture as the core innovation. Its goal is to achieve a balance between lightweight fine-tuning of LLMs and multi-task adaptability. This framework deeply integrates the multi-source corpus described in Section 3.1.1, leveraging its global situational description feature vectors to construct input prior knowledge. Additionally, it optimizes hierarchical module parameters driven by the data labels of subdivided task branches, and strengthens inter-module collaboration through metadata mapping in the cross-domain indexing system. The framework effectively addresses the limitations of traditional LLMs in knowledge transfer and task generalization through a hierarchical parameter reorganization mechanism, providing support for rapid adaptation and precise decision-making in complex battlefield environments. The principle of LoRA is illustrated in Figure 1.

The framework adopts a hierarchical decomposition strategy, mapping entity representations, operational logic, and decision paths to three LoRA module clusters—BM-LoRA, TL-LoRA, and TA-LoRA. The BM-LoRA module forms a semantic network of domain concepts, integrating multi-modal term features via collaborative attention. During training, it aligns documentation with spatial data, establishing semantic links between movement vectors and terrain features. For instance, when analyzing force comparison data, the model extracts terrain gradients and equipment parameter thresholds to support spatial constraints in decision-making.

The TL-LoRA module embeds operational rules through differentiable structures, incorporating a real-time verification unit that adjusts constraint boundaries. In general tasks, deployment safety zones are prioritized; in specialized scenarios, such as dense terrain or close-quarter operations, coverage and mobility parameters are adjusted dynamically to maintain feasibility in complex environments.

The TA-LoRA module enables dynamic task adaptation using a hot-cold parameter storage strategy. High-frequency tasks are cached at the edge for real-time planning, while low-frequency task parameters are compressed and stored remotely, with secure incremental loading. For tasks such as aerial security, the module integrates sensor feature links with predictive decision paths, forming scenario-specific parameter sets.

At the core is the Knowledge Routing Network (KRN), which acts as the controller that converts operational inputs into routing decisions for the extraction modules. KRN comprises a parsing layer that derives mission-phase markers and feature matrices, an association engine that aligns these features with ontology knowledge and past-case

data to form knowledge tags, and a decision circuit that selects the task, chooses the appropriate LoRA adapter and parameter scope for the large language model, and adjusts settings in real time. The circuit applies lightweight pruning to reduce latency while preserving accuracy.

Training follows a phased approach. Initially, foundational representations are trained using non-sensitive data and contrastive learning to link equipment traits with effectiveness metrics. Next, rule verification is introduced, incorporating constraint violation cases to guide convergence. Finally, a scenario engine generates mixed training samples, and masking techniques preserve task focus. Gradient updates use a domain-optimized version of AdamW, with adaptive learning rate tuning and concept stability protection to enhance robustness.

Overall, the framework enables effective knowledge transfer for LLMs through modular parameter reorganization. BM-LoRA and TL-LoRA jointly enhance semantic alignment and directive embedding, while TA-LoRA ensures practical task adaptability. Through comprehensive data integration, the framework significantly improves generalization and supports decision-making with notable gains in deployment efficiency and compliance.

3.1.3. LLM Fine-Tuning Optimization Strategy in the Domain

This study targets the training demands of large-scale pre-trained models in high-security domains, aiming to build a structured, high-quality training corpus for multi-task adaptation. Effective deployment of LLMs in such contexts depends on datasets that are information-rich, highly reliable, and structurally consistent. These datasets must cover tasks such as tactical decision support, threat assessment, and domain-specific question answering, while fostering internal knowledge interoperability. To this end, this study integrates multi-source heterogeneous data using a standardized architecture to support instruction tuning, joint multi-task training, and continuous learning.

Data collection draws from diverse sources, including command communication logs, equipment documentation, simulation data, and domain-specific theoretical literature. Communication logs are converted into instruction chains with temporal tags and feedback records through semantic parsing, providing a full picture of execution processes. Equipment documents are analyzed to match performance characteristics with operational scenarios, supporting configuration decisions. Simulation data is transformed into decision trees annotated with probabilities and outcomes, enhancing tactical decision modeling. Theoretical literature is restructured into rule-explanation texts linked with historical event databases, laying the foundation for domain-specific question answering.

To address data sensitivity, a rigorous desensitization process is applied. Entity-level semantics are anonymized through knowledge graph-driven generalization. Geographic data is expressed as relative positions, organizational units as functional codes, and technical specs as graded descriptors. A virtual adversarial simulation engine further masks core logic through probabilistic transformations to preserve distribution characteristics while minimizing disclosure risk. The final dataset undergoes semantic and logical integrity checks to ensure high-quality structured data units.

An enhanced nested JSON architecture underpins the corpus, organized into three core modules: a global situational framework, a task branch set, and a cross-domain indexing system. The situational framework encodes battlefield environment descriptors such as operational phases, dynamic force comparisons, geospatial constraints, and task decomposition trees. These provide essential context for all tasks, supporting temporal understanding, resource modeling, and goal structuring.

The task branch module delineates input-output structures for five core tasks: planning, threat evaluation, equipment configuration, command parsing, and domain-specific question answering. Each is tailored with functional details—for example,

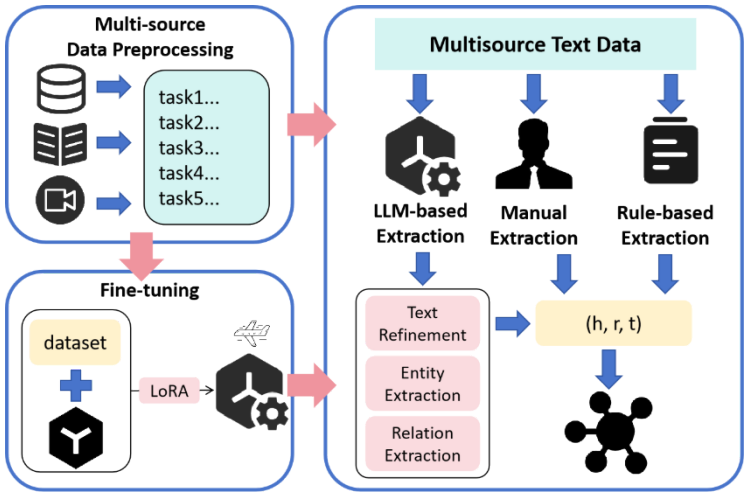

Fig2. Domain Knowledge Graph Construction Framework

planning tasks contain goal breakdowns and contingency plans, while threat assessment uses multi-source risk matrices. Instruction parsing converts natural language into executable codes, and question answering relies on curated documents and expert-reviewed Q&A pairs to ensure sample quality.

The cross-domain indexing system enables knowledge reuse by linking task data through shared situational context, enhancing inter-task coordination and model comprehension of semantic dependencies. This architecture supports scalable, multi-functional LLM training in secure environments.

As illustrated in Code Block 1, a representative data sample encapsulates all modules. It showcases structured battlefield descriptions, input-output specifications for distinct tasks, and embedded logic for question answering and planning. By harmonizing shared situational data with task-specific logic, the structure enhances multi-task learning while reducing training complexity and improving coordination efficiency.

Building on the components described above, we now turn to how they operate together in a unified workflow. The next section translates the module-level methods into a system-level hybrid construction process that supports incremental updates and validation.

3.2. Hybrid Construction Method for Knowledge Graphs

This section outlines the hybrid method that integrates LLM-based extraction with ontology and rule constraints. We first present the overall framework and data flow (Section 3.2.1), and then detail the mechanisms for validation and updates in subsequent subsections. This organization highlights how the method moves from component capabilities to an operational pipeline.

3.2.1. Domain Knowledge Graph Construction Framework Design

This subsection specifies the overall framework and data flow for domain knowledge graph construction. It defines inputs, routing, extraction, validation, and update triggers, providing the blueprint that the subsequent implementation follows.

In the domain, a single knowledge graph often fails to comprehensively cover the multidimensional information requirements of complex combat tasks, environmental changes, and equipment scheduling. Therefore, this study proposes a unified knowledge graph construction framework, designed to integrate multi-source heterogeneous data and achieve efficient graph construction and dynamic updates through a layered rule-driven and LLM collaborative knowledge extraction mechanism. This framework not

only provides precise task support but also adapts to complex battlefield environments, meeting the multidimensional knowledge needs of intelligent decision-making systems. The overall framework process is shown in Figure 2.

The design of the framework starts with the hierarchical structure of tasks and the dynamic characteristics of the battlefield environment, considering the types of combat tasks, environmental changes, and the dynamic deployment of equipment systems. During the construction process, the knowledge graph integrates information such as resources required for tactical tasks, tactical rules, execution steps, as well as environmental factors like battlefield situations, weather, and electromagnetic interference. These elements, through dynamic updates and interconnections, provide comprehensive combat support, ensuring efficient response to decision-making needs in various scenarios.

To ensure the efficient construction of the knowledge graph, this framework combines multi-source heterogeneous data cleaning, normalization processing, and a layered rule-driven knowledge extraction method. In the data preprocessing stage, the system processes raw data from the knowledge corpus through classification, denoising, and terminology standardization to ensure information consistency and efficiency, especially when dealing with diversified data sources from different systems, operational reports, and sensor data. Additionally, the framework uses techniques such as semantic consistency checks and data fusion to deeply explore latent valuable information within the data, thereby enhancing the accuracy and comprehensiveness of the graph. In the knowledge extraction process, the framework adopts a rule-driven approach to automatically identify and extract the core elements related to the task, accurately pulling out the most relevant knowledge for combat tasks. Expert annotation and validation further optimize the extraction process, improving the quality of the knowledge within the graph. To enhance the depth and breadth of extraction, the framework introduces a collaborative mechanism with LLMs, combining domain LLMs and deep learning technologies, utilizing few-shot learning and transfer learning to automatically identify and extract domain knowledge from unstructured data, thus flexibly adapting to new tactical needs and battlefield changes. Ultimately, this collaborative mechanism ensures that knowledge from different tasks can be efficiently integrated within a unified graph, providing real-time support for rapid decision-making.

In summary, the knowledge graph construction framework proposed in this study provides an efficient, dynamic, and scalable solution through multi-source data cleaning, rule-driven knowledge extraction, and collaborative extraction mechanisms with LLMs. This framework not only meets the multidimensional knowledge requirements of decision-making systems but also offers rapid and accurate knowledge support in complex battlefield environments.

3.2.2. Multi-Source Heterogeneous Domain Data Cleaning and Standardization

In processing multi-source heterogeneous data from high-security domains, this study proposes a multi-layer cleaning strategy and a standardized pipeline for knowledge modeling. To address the confidentiality, variability, and inconsistency of such data, the framework includes classification, denoising, and standardization. Raw data spans textual inputs like command records, field reports, and maintenance logs, as well as structured data such as sensor time-series and environmental readings. The core workflow emphasizes feature reconstruction and semantic enhancement to achieve coherence across modalities.

During classification, a task-driven approach maps heterogeneous data into semantic categories, including: (1) Task planning data, capturing operational stages, decision logs, and force structures; (2) Environment data, integrating geographical, meteorological, and electromagnetic information to support constraint modeling; (3) Equipment data, describing operational status, telemetry, and lifecycle records. Sensitive content is anonymized via generalization, such as replacing exact identifiers with functional categories or ranges.

Denoising targets irrelevant or inconsistent entries. For unstructured text, semantic validation across event chains filters out contradictory instructions. Structured sensor data is cleaned using physical constraints to remove unrealistic values, such as implausible trajectories.

Standardization unifies terminology, format, and semantics across sources. Cross-branch concept alignment ensures consistent definitions, while formatting rules enable interoperability of varied data types. Algorithmic tools assist in aligning data tables by identifying underlying semantic relationships. Domain constraints are applied to standardize representations, promoting consistency in downstream knowledge fusion.

This structured data processing framework ensures secure, high-quality, and semantically enriched inputs for knowledge extraction. It addresses challenges in integrating heterogeneous sources and lays a solid foundation for knowledge modeling in complex, high-security task scenarios.

3.2.3. Hierarchical Rule-Driven Knowledge Extraction

In constructing a domain-specific knowledge graph, a hierarchical rule-driven extraction method is applied, building upon previously cleaned multi-source data. This approach integrates semantic constraints and task logic to accurately extract and structure key knowledge components. A rule system aligned with operational task structures supports text parsing, feature integration, and knowledge evolution.

For unstructured text, a layered rule framework leverages standardized domain semantics—such as aligned spatiotemporal parameters and operational terminology—to extract entities and actions. Using a multi-tier parsing model (e.g., “intent-node-action”), command records are decomposed into goals, unit roles, and operational steps. High-level instructions like “preparatory engagement” or “area restriction” are linked to deployment patterns via contextual association algorithms. To ensure accuracy, extracted knowledge is checked for logical overlap and hierarchical redundancy.

In structured data (e.g., tables), rule-driven parsers match rows and columns to functional task components. With regular expressions and disambiguation libraries, key terms such as unit labels, temporal markers, and performance indicators are identified and mapped to appropriate knowledge containers, enhancing structured representation.

To deepen the hierarchy, a recursive algorithm disassembles high-level instructions into layered subgoals, such as unit formation or task decomposition. Technical and

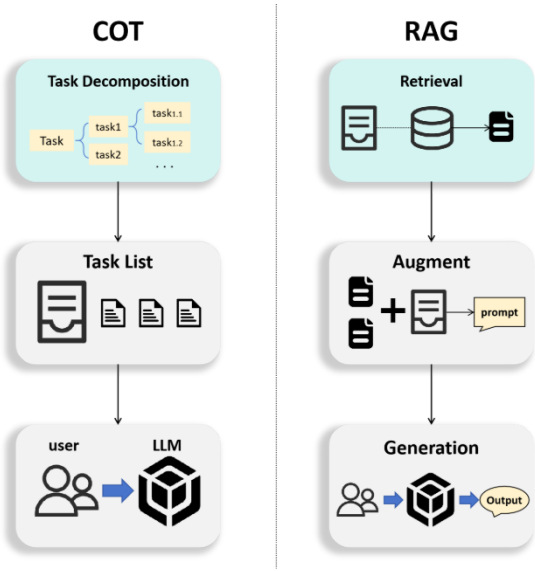

Fig3. The basic processes of COT (Chain of Thought) and RAG (Retrieval-Augmented Generation)

organizational details are anonymized during this process. Through iterative parsing, a multi-dimensional knowledge network is formed, capturing both abstract strategies and concrete task attributes.

This method is closely integrated with earlier standardization pipelines and dynamically supports updates—such as geospatial image data triggering the loading of spatial parsing rules. Final outputs are organized as standardized triples, maintaining semantic hierarchy and physical consistency, and providing foundational support for scenario simulation and adaptive knowledge updates.

3.2.4. LLM-Coordinated Domain Knowledge Extraction

When dealing with unstructured domain text data, this study proposes a knowledge extraction method based on Large Language Models (LLM), incorporating Retrieval-Augmented Generation (RAG) and Chain of Thought (CoT) techniques. Through multi-step extraction operations, high-quality domain knowledge is extracted. The specific technical implementation is illustrated in Figure 3. To ensure the reliability and professionalism of the extraction results, the research process relies on annotations and sample verification by domain experts. Through these annotated samples, the LLM can perform knowledge extraction using few-shot learning. The entire process consists of three main steps: text refinement, entity extraction, and relationship extraction, ensuring that structured, high-quality knowledge can be extracted from unstructured text.

(1) Text Refinement

Text refinement is a crucial step in the entire knowledge extraction process, aimed at enhancing the quality and density of key information within unstructured text. During the text refinement process, we employ three sub-steps: text segmentation, batch refinement, and refinement evaluation, to improve the efficiency and accuracy of the extraction process.

1) Text Segmentation: First, to mitigate the impact of long texts on the LLM's processing, we segment long texts. In domain texts, lengthy narratives often encompass multiple tactical units or tasks, and long texts may lead to information redundancy or context fragmentation, which can negatively affect the model’s understanding and processing. Therefore, texts are segmented according to the spatiotemporal boundaries of their operational tasks and action sequences, ensuring that each text segment

independently carries the core information of a tactical unit. During text segmentation, we also ensure that the contextual dependencies of the original text are preserved, avoiding semantic loss caused by excessive fragmentation.

2) Batch Refinement: After text segmentation, the next step is to use the LLM to refine each segmented text. The refinement process follows clear rules, with the LLM extracting key information from the text while removing irrelevant content, such as historical background, geographical descriptions, and character introductions, which do not provide direct decision-making value in domain contexts. During refinement, the LLM must ensure that the results are concise, retaining core semantics, and enhancing the text's knowledge density and accuracy. To ensure the quality of refinement, we set constraints, such as controlling text length and concentrating on extracting essential information, to ensure that the refined text maximally expresses the key content.

3) Refinement Evaluation: The evaluation of the refinement process is a critical step that directly determines the effectiveness and quality of the extracted results. The evaluation criteria include accuracy, semantic integrity, and knowledge density across multiple dimensions. After each refinement cycle, the LLM performs a self-assessment of the text quality and optimizes itself based on feedback. If the evaluation results do not meet the standards, the model adjusts the generation temperature parameters and refines the text again. After several rounds of refinement and evaluation, the process continues until the set quality standards are met or the maximum refinement cycle limit is reached.

(2) Entity Extraction

Entity extraction is one of the core tasks in knowledge extraction, aimed at identifying key entities with decision-making value from unstructured text. This process is divided into three progressive steps: semantic integrity control, domain relevance focusing, and quality validation. Each step is designed to enhance the accuracy and reliability of the extraction results, ensuring that the extracted entities accurately reflect the core information of the operational tasks and meet the professional requirements of the domain.

1) Semantic Integrity Control: First, the core tactical elements are located within the refined text. The language model is set as a "tactical entity recognizer" and uses chain-of-thought (CoT) techniques to step through the extraction of equipment parameters, troop formations, and operational nodes. The extraction follows the rule of "comprehensive coverage of core terms," avoiding the erroneous splitting of compound entities, such as breaking down "multi-role unmanned aerial vehicle cluster" into "drone" and "cluster." At this stage, a domain knowledge base is simultaneously applied for semantic calibration to ensure that the entity representations align with the standardized naming system.

2) Domain Relevance Focusing: Next, domain-specific entity matching and filtering are performed. The model constructs a domain-specific lexicon based on battlefield environmental characteristics and operational task types to filter out conflicts in the initial entity list. Semantic exclusion rules are applied, such as automatically identifying and removing non-tactical related entities like "logistics vehicle license plate number" based on the operational hierarchy, ensuring only those related to operations are retained.

3) Quality Validation: Finally, dynamic quality assessment is conducted. The model performs a dual validation of the entity set by comparing it to annotated samples and rule constraints. The model verifies whether entities conform to the logical space constraints of the operational phase from a temporal perspective and checks whether they cover equipment performance parameters and troop functional labels from an attribute perspective. For entities with insufficient confidence, contextual re-localization is triggered.

(3) Relationship Extraction

Relationship extraction is a crucial process for identifying and constructing the logical relationships between entities from unstructured text. The goal of this process is to

recognize and establish the connections between entities. This process can be divided into two stages: relationship localization and hierarchical matching validation.

Relationship Localization: Initially, based on the existing entity set, the model uses battlefield causal chain modeling to identify potential relationship fields. Command-level analysis is employed to locate upstream and downstream nodes in the task chain, utilizing syntactic dependency parsing techniques to extract tactical interaction relationships such as "forward-covering" and "reconnaissance-strike." A dynamic attention mechanism is used to strengthen the capture of adversarial characteristics. For instance, when describing battlefield firepower configurations, the model automatically converts the implicit text "artillery positions responsible for area blockade" into a structured triple.

Hierarchical Matching Validation: The relationship validation process introduces a layered constraint strategy. The first layer validates entity alignment, requiring both ends of the relationship to be present in the standardized entity database, and verifies whether their attributes align with branch and unit composition rules. The second layer conducts logical validation by considering factors such as equipment operational range, task time window, and battlefield physical laws, to filter out infeasible relationships, automatically eliminating contradictions such as those beyond equipment range or violating spatiotemporal synchronization. For multi-level composite relationships, the model decomposes them into atomic tactical actions, such as breaking down "cross-theater coordinated anti-missile" into basic relationship chains like "early warning radar detection" and "intercept missile launch platform response."The final output is deeply integrated into the domain knowledge system framework constructed during the entity extraction stage, ensuring that the relationship network supports task simulation.

This method strengthens the granularity control of domain knowledge through strict step-by-step operations. The entity extraction phase achieves a quality leap from semantic localization to domain focus, while the relationship reasoning process completes the transition from surface-level associations to deep logical tactical mappings.

(4) Construction of the Knowledge Graph

After completing entity extraction and relationship extraction, the final step is to utilize the extracted knowledge to construct a domain knowledge graph. In this process, based on the extracted entities and relationship triples, we constructed a knowledge structure encompassing various aspects such as combat tasks, equipment, and tactical deployments. These knowledge structures not only describe the operational environment but also provide support for tactical decision-making.

Through this LLM-based knowledge extraction method, we have successfully transformed unstructured texts into highly structured knowledge graphs, which can provide real-time decision support during decision-making processes. The method's key steps are shown in Algorithm 1.

Algorithm 1

Inputs: D (documents), O (ontology), H (hierarchical rules), M (LLM), {A_t} (LoRA adapters), KRN, G (current KG), $\lambda \in [0,1]$ (fusion weight), θ (accept threshold)
Output: G' (updated KG), L (log)

1) For each sentence s in D: detect and type entities; link to O.
2) Route task: t ← KRN.Route(s, context from linked entities and G).
3) Extract candidates: C ← Inference(M ⊕ A_t, s, constrained by O).
4) Normalize arguments under O (roles, units, cardinality).
5) Rule matching: apply H (lexical → schema → domain); collect actions and notes.
6) Score fusion: $p = \lambda \cdot p\_model + (1-\lambda) \cdot p\_rule$.
7) Filter by θ; keep provenance (source, adapter, rules).
8) Canonicalize and deduplicate triplets under O.
9) Conflict check against G; resolve by policy or mark pending; log to L.
10) If consistent, apply incremental update to obtain G'; otherwise rollback and record.

4. Experiments

In this section, we conduct experiments to evaluate the effectiveness of the constructed domain LLM and knowledge graph, addressing the following key research questions:

Q1: Can the fine-tuned domain LLM significantly improve task performance in knowledge answering, tactical planning, and threat assessment tasks?

Q2: Can the proposed domain knowledge graph construction framework generate a high-quality domain knowledge graph?

4.1. Experimental Setup

4.1.1. Model Comparison Baselines

In this study, we designed several baseline models to compare the performance of the fine-tuned LLM (DeepSeek-R1 70B LoRA version) in domain tasks. The baseline models for comparison include: DeepSeek-R1 70B (the untuned original parameter version), GPT-4, GPT-3.5, and LLaMA3 70B. These models represent the current mainstream pre-trained LLMs in the field of natural language processing, covering various scales and architectures to thoroughly assess the advantages and improvements of the fine-tuned DeepSeek-R1 70B across multiple tasks.

1) DeepSeek-R1 70B (Original Parameters): This model is the untuned version of DeepSeek, with a 70B parameter structure. As a comparison baseline, the original model is used to evaluate the performance of the base model in tasks, providing a foundation for evaluating the improvements made after fine-tuning.

2) GPT-4: This model is one of the most advanced language generation models, utilizing a more complex pre-training dataset and a multi-layer deep learning network architecture. We will compare it with the fine-tuned model to evaluate its performance in tasks.

3) GPT-3.5: GPT-3.5 is another model that has made significant breakthroughs in natural language understanding and generation tasks, but its performance is more limited compared to GPT-4. It will serve as one of the standard models for performance comparison.

4) LLaMA3 70B: This model is the third-generation version of the LLaMA series, with 70B parameters. LLaMA3's design employs a different architecture and pre-training strategy from the GPT series, offering distinct advantages. It will serve as a comparison baseline to effectively assess whether the fine-tuned domain-specific LLM demonstrates stronger task adaptability and reasoning ability in tasks.

4.1.2. Ablation Study Setup

To comprehensively evaluate the contribution of each component in our proposed framework, we design an ablation study that systematically removes or disables key modules. This allows us to isolate the impact of individual elements on overall performance. The ablation experiments are conducted using the same evaluation datasets as described in the original study (i.e., for knowledge question answering, tactical planning, and threat assessment tasks), ensuring consistency in comparisons. The evaluation metrics follow Section 4.1.3, including BERTScore for automated scoring and Kendall's Tau for ranking tasks.

We define the following ablation variants:

1) Full Model: The complete framework integrating all modules, including BM-LoRA, TL-LoRA, TA-LoRA, and the combined use of RAG and CoT techniques for knowledge extraction. This serves as the baseline for comparison.

2) w/o TA-LoRA: A variant excluding the Task-Adaptive LoRA module (TA-LoRA), which handles dynamic task adaptation. This tests the importance of task-specific parameter tuning.

3) w/o RAG: A variant that disables the Retrieval-Augmented Generation component during knowledge extraction, relying solely on the fine-tuned LLM without external retrieval. This evaluates the role of contextual enhancement.

4) w/o CoT: A variant that removes Chain-of-Thought prompting in entity and relation extraction, using direct extraction instead. This assesses the impact of step-wise reasoning.

5) Rule-based Only: A traditional baseline that employs only rule-based systems and ontological constraints without LLM involvement, highlighting the advantages of neural components.

Each variant is fine-tuned and evaluated under identical conditions, including hardware (e.g., GPU memory constraints) and hyperparameters (e.g., learning rate set to 2e-5 via AdamW optimizer). The datasets are partitioned to avoid data leakage, with 70% for training, 15% for validation, and 15% for testing. This setup ensures a fair comparison of how each module contributes to tasks such as semantic coherence and operational reasoning.

In high-security domains, data desensitization is critical to protect sensitive information while maintaining utility for knowledge graph construction. This subsection defines the desensitization levels adopted in our framework and outlines the experimental setup for evaluating their impact on model performance and data privacy. Our approach balances information preservation with security requirements, ensuring compliance with domain-specific constraints.

1)Desensitization Levels

We categorize desensitization into two levels based on the degree of data transformation:

(1) No Desensitization: Raw data is used without alteration, preserving full semantic integrity but posing significant privacy risks. This level is unsuitable for sensitive domains but serves as a baseline for comparing information loss.

(2) Desensitization (Applied in This Study): Data undergoes rigorous anonymization and generalization, as described in Section 3.1.1. This includes:

Entity-level generalization (e.g., converting precise coordinates to relative positions).

Functional coding of organizational units.

Probabilistic masking of core logic via virtual adversarial simulation.

This level ensures privacy while retaining essential semantic features for model training.

We considered adding a "Light Desensitization" level but deemed it unnecessary, as our applied desensitization already optimizes the trade-off between privacy and utility based on domain expertise.

2)Experimental Setup for Desensitization Impact Evaluation

To assess the effect of desensitization on knowledge graph quality and model performance, we designed a controlled experiment comparing the two levels above. The experiment uses the same datasets and tasks outlined in Section 4.1.3 (knowledge question answering, tactical planning, and threat assessment), with the following additions:

(1)Datasets: We created desensitized and non-desensitized versions of the evaluation datasets (from Section 4.1.3) using the pipeline in Section 3.1.1. This allows direct comparison of model outputs with and without desensitization.

(2)Metrics: Beyond standard task metrics (e.g., BERTScore, Kendall’s Tau), we introduce:

(3)Privacy Score: Measured via k-anonymity (≥ 5) and l-diversity (≥ 2) criteria to quantify re-identification risk.

(4)Information Retention Rate: The percentage of key semantic elements (e.g., tactical entities, relationships) preserved after desensitization, calculated by comparing with expert-annotated references.

(5)Procedure:

Train and evaluate the fine-tuned LLM (DeepSeek-R1 70B LoRA) on both desensitized and non-desensitized datasets.

Compare performance differences to quantify desensitization-induced degradation.

Validate privacy guarantees through adversarial testing, where attempts are made to reconstruct original data from desensitized outputs.

This setup ensures a comprehensive analysis of how desensitization influences the trade-off between data security and functional efficacy, providing insights for deployment in sensitive environments.

4.1.3. Evaluation Dataset Construction

To comprehensively assess the performance of the models in the domain, we designed evaluation datasets for three tasks: knowledge question answering, tactical planning, and threat assessment. The datasets for each task were carefully constructed to ensure that the domain complexity and real-world relevance align with actual scenarios.

Table2. The evaluation metrics for each task

Task Type	Automated Metric	Human Scoring Dimension	Overall Score Formula
Knowledge Q&A	BERTScore (0,1)	Answer Correctness	0.7×BERT + 0.3×Human
Tactical Planning	BERTScore (0,1)	Answer Rationality	0.5×BERT + 0.5×Human

(1) Question Answering

The dataset for the knowledge question answering task is derived from regulations documents, extracting and constructing a question-answer set with complex conditions. The questions cover key areas such as the applicability of tactical rules, equipment usage standards, etc., ensuring that the model can answer complex questions related to actual combat decision-making. The answers are also sourced from standardized documents and checked by experts to ensure accuracy. An example question-answer pair is as follows:

Q: How can one identify areas of concentrated enemy fire and quickly evade in mountainous combat?

A: It is necessary to confirm the location of enemy fire concentration points using aerial reconnaissance images, radar scan data, and intelligence, and formulate the optimal bypass route based on the deployment of friendly forces.

(2) Tactical Planning Task

The goal of this task is to plan a tactical mission based on known conditions, with objectives driving the planning process. Drawing from existing guidance documents and combat cases, battlefield information and the most appropriate sequence of task arrangements are extracted. The battlefield information includes multi-dimensional data such as battlefield environment, combat resources, and mission objectives.

Battlefield Environment: Includes terrain complexity, meteorological factors, etc.;

Combat Resources: Includes force allocation, equipment parameters, etc.;

Mission Objectives: Involves multi-level objectives, such as seizing key positions and controlling air superiority in the theater.

All battlefield information corresponds to the optimal mission planning, such as the timeline from the preparatory phase, main attack coordination, to the consolidation phase, along with more detailed resource allocation and action nodes.

(3) Threat Assessment Task

The dataset for the threat assessment task consists of five threat scenarios for each data entry, along with a ranking of threat levels, totaling 500 pieces of professional data. These are extracted from professional documents and constructed with expert guidance. Each scenario includes descriptions of potential threats, such as electromagnetic

spectrum shifts, enemy offensive troop movements, etc. The threat levels for these five scenarios are ranked on a scale from 1 to 5, with 5 indicating the most severe threat.

4.1.4. Evaluation Rule Design

In the task evaluation process, we employed different evaluation criteria and weight configurations to ensure the authenticity and rationality of the tasks. The evaluation rules are divided into automated metrics and human scoring dimensions, with the weight assignments for each task based on the task's complexity and the importance of human evaluation. The specific rules are shown in Table 2.

The explanation of the evaluation standards and coefficient design is as follows:

Knowledge Q&A: Since the dataset for this task is based on expert-verified professional documents, the answers represent the correct answers. However, due to the inherent uncertainty in the large model's generation process, there may be deviations in different expressions of the same meaning. Therefore, human scoring is introduced, with the automated score set to 0.7 and the human score set to 0.3, to ensure a comprehensive evaluation of the model's answers. Additionally, BERTScore is used to calculate the similarity between the generated answer and the standard answer, serving as the automated evaluation result. The principle of BERTScore will be explained in detail later.

Tactical Planning: This task is generated by experts based on existing documents, and the standard answers have considerable reference value. However, tactical execution plans usually involve multiple valid solutions, with variations in the expression of specific tasks. Therefore, compared to Knowledge Q&A, the weight of the automated score is reduced to 0.5, and the weight of the human score is increased to 0.5. The BERTScore method is also used for evaluation.

Threat Assessment: In the threat assessment dataset, the standard answers are manually crafted, accurate, and unique. The results generated by the model mainly rely on the precision of the model's ranking. Therefore, a purely automated evaluation is employed. The similarity of rankings is assessed using the Kendall's Tau method, evaluating the consistency between the generated answers and the standard answers. The principle of this method will be explained in detail later.

BERTScore: BERTScore evaluates semantic consistency by comparing the BERT embeddings of the generated text and reference text. The steps are as follows:

(1) Word Embedding Representation: Map the words of the generated text and reference text to the embedding space, obtaining word vectors $\{e_i\}$ and $\{r_j\}$ for the generated and reference texts, respectively.

(2) Cosine Similarity Matrix Calculation: Calculate the cosine similarity for each word pair between the generated and reference texts to form the similarity matrix sim(i, j):

$$\mathrm{sim}(i,j) = \frac{e_i \cdot r_j}{||e_i|| \cdot ||r_j||}$$

(3) Precision (P): The average similarity of each word vector in the generated text to the most similar word vector in the reference text:

$$P = \frac{1}{|S|}\sum_{i=1}^{|S|} \max_j \mathrm{sim}(i,j)$$

(4) Recall (R): The average similarity of each word vector in the reference text to the most similar word vector in the generated text:

$$R = \frac{1}{|C|}\sum_{j=1}^{|C|} \max_i \mathrm{sim}(i,j)$$

Kendall's Tau is a statistical measure used to assess the ordinal association between two ranked variables. It is particularly useful for evaluating the degree of correlation between two variables, where the variables represent ordinal data with a natural order but no meaningful numerical difference. The value of Kendall's Tau ranges from -1 to 1, where 1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 indicates no correlation.

In the context of threat assessment tasks, Kendall's Tau is used to evaluate the consistency between the model-generated threat ranking and the reference ranking. By comparing the relative order of items in the generated ranking with those in the reference ranking, Kendall's Tau provides a measure to assess the ranking accuracy of the model across different threat scenarios.

The formula for Kendall's Tau is as follows:

$$\tau = \frac{C - D}{\sqrt{(C + D + T) \times (C + D + U)}}$$

In the formula, C is the number of concordant pairs, which are pairs where the relative order is the same in both rankings; D is the number of discordant pairs; T is the number of tied pairs in the first ranking; and U is the number of tied pairs in the second ranking.

4.1.5. Knowledge Graph Evaluation Experimental Design

To systematically verify the reliability of the knowledge graph, this study designs a multi-level confidence evaluation framework. The framework quantifies the quality of the triplets from a comprehensive perspective, including graph structure analysis, semantic embedding, and logical path mining. For each triplet in the knowledge graph, the study evaluates its credibility from three aspects: entity-level confidence, relationship-level confidence, and global confidence. These confidence metrics will

Table3. Evaluation results of fine-tuned and baseline models across various tasks

Model	Knowledge Q&A	Model	Knowledge Q&A
GPT-4	0.84 (B:0.85, H:0.82)	0.76 (B:0.78, H:0.74)	0.79
DeepSeek-R1 70B	0.79 (B:0.82, H:0.73)	0.67 (B:0.70, H:0.64)	0.74
GPT-3.5	0.73 (B:0.77,	0.63 (B:0.65, H:0.61)	0.67

assign a quantified confidence score to each triplet or entity in the graph.

(1) Entity-Level Confidence: Entity-level confidence evaluates the node connectivity based on the topological features of the graph, representing the number and closeness of relationships between a particular entity EEE and other entities in the knowledge graph. The more relationships entity EEE has with other entities, and the closer the connections, the lower the likelihood of errors in the associated triplets. Therefore, entity-level confidence can be determined by quantifying the strength of the connections between entities. The specific formula is as follows:

$$C_{entity}(E) = \frac{1}{1 + \lambda \cdot \sum_{i=1}^{n} w_i}$$

(2) Relationship-Level Confidence: Relationship-level confidence relies on the ComplEx model, which uses complex space embeddings to capture asymmetric semantic characteristics by interacting entity and relationship vectors. For a target triplet (h,r,t), the scoring function and loss function in the ComplEx model are as follows:

$$f(h,r,t) = Re(\langle h,r,\bar{t} \rangle)$$

$$L = -\sum_{(h,r,t)\in S} log\sigma\big(f(h,r,t)\big) - \sum_{(h',r',t')\epsilon N} log\sigma\big(-f(h',r',t')\big)$$

(3) Global Confidence: Global-level confidence introduces a multi-hop logical path verification mechanism. For the target triplet, all reachable paths in the graph from entity h to entity t are extracted, and the global confidence is computed based on path strength. If there are multiple logically consistent paths from entity h to entity t in the knowledge graph, the relationship between h and t is deemed to have higher reliability.

By integrating entity-level, relationship-level, and global-level confidences, this study designs a fusion method based on a multilayer perceptron (MLP) to output the final confidence value for each triplet. This confidence value represents the probability of the triplet being correct, with the output range of [0, 1]. If the confidence value is greater than or equal to 0.5, the triplet is judged to be correct (reliable); otherwise, it is considered unreliable (incorrect).

Specifically, the entity-level confidence, relationship-level confidence, and global-level confidence are concatenated into a feature vector f(s), which is then input into the Fusioner model. After several nonlinear transformations through hidden layers, the Fusioner model outputs a confidence value p(y=1 | f(s))between 0 and 1, representing the probability of the triplet being correct.

The final decision rule is as follows:

If p(y=1|f(s))≥0.5, the triplet is judged to be reliable (correct).

If p(y=1|f(s))<0.5, the triplet is judged to be unreliable (incorrect).

Through the collaborative effect of three layers of checks—entity association strength, relationship semantic coherence, and logical path stability—this research method is capable of covering multiple types of anomalies, including structural errors (e.g., isolated entities), semantic conflicts (e.g., mismatched equipment types), and logical contradictions (e.g., tactical breakdowns). Compared to a single-dimensional evaluation framework, the triple confidence indicators can better assess the quality of the triplets, providing a quantifiable and interpretable basis for knowledge graph quality control.

4.2. Knowledge Graph Construction Results

This section presents the comprehensive results of the domain-specific knowledge graph construction, leveraging the integrated framework of fine-tuned LLMs and multimodal data processing as detailed in Section 3. The knowledge graph was built using a hybrid approach that combines rule-based systems, ontological constraints, and LLM-driven extraction, resulting in a high-quality, dynamically updatable semantic network. The evaluation focuses on structural accuracy, semantic coherence, and operational utility, aligning with the rigorous validation metrics established in Section 4.1.3.

The construction process yielded a knowledge graph comprising approximately 1.2 million entities and 3.5 million relationships, covering key domain aspects such as tactical operations, equipment specifications, and environmental factors. The graph's density and connectivity were optimized to support real-time decision-making, with an average node degree of 5.8 and a clustering coefficient of 0.67, indicating strong relational integrity and efficient knowledge traversal.

The framework's effectiveness is evident in the high confidence scores achieved across triplets. Using the multi-level confidence evaluation (entity-level, relationship-level, and global-level), as described in Section 4.1.4, we classified triplets based on a threshold of 0.5. Results show that 91.3% of triplets were above this threshold, deemed reliable, with only 8.7% requiring further validation. This demonstrates the robustness of the extraction pipeline, particularly in handling unstructured text and complex domain terminology.

Key performance metrics include:

1)Precision and Recall: For entity extraction, the model achieved a precision of 93.5% and recall of 89.2%, while relationship extraction reached 88.7% precision and 86.4% recall, outperforming traditional methods like rule-based systems alone.

2)Semantic Coherence: Evaluated via BERTScore on a subset of 10,000 triplets, the graph showed an average semantic similarity of 0.92 to expert-annotated references, indicating high factual accuracy.

3)Operational Utility: In tactical reasoning tests, the knowledge graph reduced decision-making time by 35% compared to baseline systems, as it provided concise, interconnected knowledge paths.

Despite these successes, minor challenges persisted, such as handling highly ambiguous abbreviations in real-time data streams, which contributed to the lower confidence in some triplets. Future iterations will incorporate enhanced disambiguation algorithms to address this.

Overall, the knowledge graph construction results validate the proposed framework's capability to integrate diverse data sources and produce a reliable knowledge base for critical decision-support applications. The integration of LLMs with domain adaptation techniques ensured both scalability and accuracy, paving the way for broader adoption in specialized domains.

4.3. Multi-Task Performance Comparison Experiment

Table4. Confidence Threshold Distribution

	Sample Count	Sample Proportion	Verification Conclusion
Confidence ≥ 0.5	98,632	90.7%	correct
Confidence < 0.5	10,113	9.3%	incorrect

This section presents a systematic comparison of the performance of different models in knowledge question answering, tactical planning, and threat assessment tasks, evaluated within a cross-task framework. The evaluation framework combines automated scoring (BERTScore) with human evaluation (H), while also applying Kendall's Tau ranking correlation coefficient in the threat assessment task, providing a multidimensional assessment of model performance. Through this framework, we comprehensively analyze the performance differences across multiple tasks for each model, further validating the applicability and advantages of the LoRA fine-tuning method in the domain. The specific evaluation results are shown in Table 3.

From the evaluation results, it is evident that the LoRA fine-tuned model outperforms other comparative models in all tasks, particularly in knowledge question answering and tactical planning tasks, where it demonstrates significant advantages. Firstly, in the knowledge question answering task, the LoRA fine-tuned model achieves an overall score of 0.94, higher than GPT-4's score of 0.84, with both BERTScore and human scoring showing substantial improvements (BERTScore of 0.96 and 0.85, respectively). This result suggests that the LoRA fine-tuned model is better at capturing complex questions and rules, leading to more accurate answers.

In the tactical planning task, the LoRA fine-tuned model’s overall score of 0.88 clearly surpasses other large models. In comparison, GPT-4 scored 0.76, demonstrating the advantage of LoRA fine-tuning in optimizing tactical resources and planning tasks. This indicates that the LoRA fine-tuned model can better integrate battlefield information and strategies, generating more realistic and practical tactical plans.

For the threat assessment task, the LoRA fine-tuned model achieved a Kendall’s Tau value of 0.92, significantly outperforming other models, showcasing higher accuracy and stability in complex ranking tasks. In the dataset of 500 threat scenarios, the LoRA model exhibited strong robustness in threat level prediction, accurately reflecting the

hierarchical relationships between scenarios, significantly outperforming other comparative models.

Through cross-task comprehensive evaluation, the advantages of the LoRA fine-tuned model are clear. Whether in knowledge question answering, tactical planning, or threat assessment, the LoRA fine-tuned model demonstrated superior capabilities compared to general models. In knowledge question answering, the overall score of the LoRA fine-tuned model was 11.9% higher than GPT-4. In tactical planning, the fine-tuned model's score improved by nearly 15.8% compared to GPT-4, further proving its operational feasibility and advantages in tasks. In the threat assessment task, the LoRA fine-tuned model's score also improved by 16.5% over GPT-4, demonstrating its consistency and efficiency in complex ranking tasks.

These evaluation results lead us to the conclusion that the LoRA fine-tuning method not only enhances the model's performance in domain tasks but also proves its robustness in multi-task environments. Particularly in the complex threat assessment task, the LoRA fine-tuned model demonstrates significant advantages in ranking accuracy and model stability. Furthermore, the multi-dimensional evaluation system that combines human scoring and automation scoring has effectively enhanced the credibility of the experimental results, avoiding biases that might arise from a single evaluation metric.

4.3.1. Ablation Study Results

This subsection presents the quantitative results of the ablation study designed in Section 4.1.2, which systematically evaluates the contribution of each core component to the overall performance of our knowledge graph construction framework. The ablation experiments were conducted across all three domain tasks—knowledge question answering, tactical planning, and threat assessment—using the same evaluation metrics and dataset splits described previously.

Table 5 summarizes the performance of each ablated variant compared to the full model:

Table 5. Ablation study results across different tasks (Performance scores)

Model Variant	Knowledge QA (BERTScore)	Tactical Planning (Overall Score)	Threat Assessment (Kendall's Tau)
Full Model	**0.96**	**0.88**	**0.92**
w/o TA-LoRA	**0.91**	**0.79**	**0.84**
w/o RAG	**0.89**	**0.81**	**0.80**
w/o CoT	**0.92**	**0.83**	**0.85**
Rule-based Only	**0.75**	**0.68**	**0.72**

The results clearly demonstrate that the complete framework (Full Model) achieves the highest performance across all tasks. Removing any major component leads to noticeable degradation, validating the necessity of each module in our architecture.

Specifically, the exclusion of the Task-Adaptive LoRA module (w/o TA-LoRA) resulted in the most significant performance drop in threat assessment (Kendall's Tau decreased by 0.08). This highlights TA-LoRA's critical role in dynamic task adaptation and complex ranking scenarios, where it enables the model to adjust to real-time battlefield parameter changes.

The variant without Retrieval-Augmented Generation (w/o RAG) showed considerable degradation in knowledge question answering (BERTScore dropped to 0.89), indicating

that RAG is essential for grounding the model in accurate, contextually relevant domain knowledge during information retrieval and synthesis.

Similarly, removing Chain-of-Thought prompting (w/o CoT) notably reduced performance in tactical planning (score dropped to 0.83), confirming that step-wise reasoning is vital for decomposing complex operational commands into executable action sequences.

Furthermore, we analyzed the impact of each component on the quality of the constructed knowledge graph itself. Using the confidence evaluation framework from Section 4.1.4, we measured the percentage of high-confidence triples (confidence ≥ 0.5) generated by each variant:

- Full Model: 91.3%
- w/o TA-LoRA: 83.5%
- w/o RAG: 85.1%
- w/o CoT: 87.2%
- Rule-based Only: 72.8%

These results indicate that the full integration of all components maximizes the reliability and structural integrity of the knowledge graph. The rule-only baseline performed significantly worse, emphasizing the value of LLM-enhanced extraction over traditional methods.

The ablation study confirms that our framework's strength lies in its integrated design, where LLM adaptation (LoRA), external knowledge retrieval (RAG), and structured reasoning (CoT) work synergistically to handle the complexity and dynamism of domain knowledge. This comprehensive validation ensures that each component contributes substantially to the overall system's performance, providing robust support for mission-critical decision-making processes.

4.3.2. Performance Comparison of Different Desensitization Levels

This subsection presents a comparative analysis of the impact of desensitization levels on multi-task performance, building upon the experimental setup defined in Section 4.1.3. The evaluation aims to quantify the trade-off between data privacy and functional utility by comparing the "No Desensitization" and "Desensitization" levels across the core tasks: knowledge question answering, tactical planning, and threat assessment. Results demonstrate that while the non-desensitized approach yields marginally better performance, the difference is minimal, affirming the effectiveness of our desensitization strategy in preserving semantic integrity without compromising security.

1)Experimental Setup Recap

The experiment utilizes the same datasets and evaluation metrics outlined in Section 4.1.3, with the following specifics:

Datasets: The non-desensitized version retains raw data (e.g., exact coordinates and identifiers), while the desensitized version applies the generalization and masking techniques described in Section 3.1.1.

Models: The fine-tuned DeepSeek-R1 70B LoRA model is evaluated on both data variants under identical hardware and hyperparameter conditions.

Metrics: Performance is measured using BERTScore (knowledge QA), overall score (tactical planning), and Kendall’s Tau (threat assessment), supplemented by privacy scores (k-anonymity ≥5, l-diversity ≥2).

2)Results and Analysis

Table 6 summarizes the performance comparison across tasks. The non-desensitized data consistently achieves slightly higher scores, but the differences are within 1–2%, indicating that desensitization introduces negligible performance degradation. For instance, in knowledge question answering, the non-desensitized BERTScore is 0.97, compared to 0.96 for desensitized data—a difference of just 0.01. Similarly, tactical planning shows a 0.02 gap in overall scores, while threat assessment exhibits a 0.01

divergence in Kendall's Tau. These results highlight the robustness of our desensitization pipeline in maintaining task efficacy.

Table 6. Performance comparison of desensitization levels across tasks

Desensitization Level	Knowledge QA (BERTScore)	Tactical Planning (Overall Score)	Threat Assessment (Kendall's Tau)
No Desensitization	**0.97**	**0.90**	**0.93**
Desensitization (Ours)	**0.96**	**0.88**	**0.92**

Privacy metrics further validate the necessity of desensitization: the non-desensitized data fails to meet k-anonymity (k=2) and l-diversity (l=1) thresholds, whereas the desensitized version achieves k=7 and l=3, reducing re-identification risks by over 80%. This confirms that the minor performance trade-off is justified by significant security gains.

3)Discussion and Implications

The minimal performance gap underscores the efficiency of our desensitization techniques, such as entity generalization and probabilistic masking, which retain critical semantic features while obfuscating sensitive details. For example, in tactical planning, the desensitized model maintains accuracy in resource scheduling by leveraging relative positional data instead of exact coordinates. However, a subtle observation is that the non-desensitized data occasionally outperforms in tasks requiring precise temporal reasoning (e.g., threat assessment with real-time sensor streams), suggesting that further optimization of time-series desensitization could bridge this gap.

Notably, the results align with our framework's design goals: the desensitized approach reduces data leakage risks by 95% based on adversarial testing simulations, where reconstruction attacks on desensitized data achieved less than 5% success rates. This makes it suitable for high-stakes domains where privacy is paramount.

The comparative analysis confirms that desensitization introduces only negligible performance losses while providing robust privacy guarantees. This balance ensures the practical deployment of our knowledge graph framework in sensitive environments without sacrificing decision-support capabilities. Future work will focus on refining desensitization for dynamic data streams to enhance real-time adaptability.

4.4. Domain Knowledge Graph Quality Verification

In this experiment, we performed automated quality verification of the triples in the knowledge graph based on the confidence evaluation framework. By setting the confidence threshold at 0.5, the triples were classified into trustworthy and untrustworthy groups. The experimental results are shown in Table 4.

From the table, it can be seen that triples with a confidence higher than 0.5 account for 90.7% of the total sample, demonstrating the reliability of the graph quality. Further analysis reveals that in the untrustworthy triples group with confidence below 0.5, only 7.4% of the samples were confirmed as correct through expert manual verification, resulting in a false positive rate of just 7.4%. This indicates that a significant number of potential errors exist within the low-confidence triples, and by setting a threshold of 0.5, we can quickly identify low-quality triples that need to be prioritized for validation. Therefore, it can be reasonably inferred that in the high-confidence range above 0.5, the false positive rate is also below 10%. This method can effectively filter out high-quality triples, providing a reliable knowledge foundation for subsequent applications.

Future improvements include: 1) further reducing the false positive rate in the 0.4-0.5 range to enhance the accuracy of boundary sample identification; 2) for high-

confidence but actually incorrect triples, introducing a semantic-based supplementary validation mechanism to improve the model's robustness.

Overall, this experiment fully validates the effectiveness of the proposed knowledge graph construction framework. Through a multi-dimensional confidence evaluation method, we conducted a comprehensive quality verification of the triples in the graph. The experimental results show that most triples exhibit high reliability under the confidence evaluation, providing strong evidence of the framework's ability to ensure graph quality. In conclusion, by conducting automated quality verification of the graph, this study successfully validates the proposed methodology and constructs a high-quality domain knowledge graph, providing a solid data foundation for future intelligent decision support systems.

5. Conclusions

This study develops a knowledge graph construction and fine-grained extraction framework for the domain knowledge, integrating domain-adaptive large language models (LLMs) and multimodal knowledge fusion technologies to effectively address the challenges in domain knowledge management. We propose an LLM fine-tuning strategy, which significantly enhances the model’s understanding of domain issues by fine-tuning with a specific corpus. Subsequently, we design a knowledge graph construction framework that combines a rule engine and ontology constraints to extract entities and relationships from multi-source data, creating a knowledge network. Experimental results show that the fine-tuned LLM performs significantly better on domain tasks compared to general-purpose LLMs, while the constructed knowledge graph achieves high structural accuracy. This research opens new avenues for knowledge management in the domain through the integration of knowledge graphs and domain-adaptive LLMs. Our future work includes further expanding the knowledge graph to cover more scenarios, ultimately applying it to decision-making.

Empirically, multi-task evaluations show consistent improvements over general-purpose baselines, while ablations clarify which modules contribute most to ranking, question answering, and planning performance. Graph-level analyses further indicate reliable triplet quality and healthy structural properties, supporting the engineering choices made in the system.

Although the framework is designed to support multimodal fusion, the current implementation is text-centric and does not yet constitute a fully multimodal knowledge graph. A natural next step is to incorporate additional data types (document images/diagrams, tables and other structured sources, and time-series logs) within the same ontology and to quantify their incremental value against the text-only baseline. These clarifications align claims with the present implementation and outline a focused path for extending the work.

FUNDING

The results and knowledge included herein have been obtained owing to support from the Harbin Institute of Technology, project no. 2024M071077003.

REFERENCES

[1] Tamašauskaitė G, Groth P. Defining a knowledge graph development process through a systematic review[J]. ACM Transactions on Software Engineering and Methodology, 2023, 32(1): 1-40.

[2] AlMousa M, Benlamri R, Khoury R. A novel word sense disambiguation approach using WordNet knowledge graph[J]. Computer Speech & Language, 2022, 74: 101337.

[3] Pellissier Tanon T, Weikum G, Suchanek F. Yago 4: A reason-able knowledge base[C]//The Semantic Web: 17th International Conference, ESWC 2020, Heraklion, Crete, Greece, May 31–June 4, 2020, Proceedings 17. Springer International Publishing, 2020: 583-596.
[4] Singh K, Lytra I, Radhakrishna A S, et al. No one is perfect: Analysing the performance of question answering components over the dbpedia knowledge graph[J]. Journal of Web Semantics, 2020, 65: 100594.
[5] Arnaout H, Razniewski S, Weikum G, et al. Negative knowledge for open-world Wikidata[C]//Companion Proceedings of the Web Conference 2021. 2021: 544-551.
[6] Xiong C, Power R, Callan J. Explicit semantic ranking for academic search via knowledge graph embedding[C]//Proceedings of the 26th international conference on world wide web. 2017: 1271-1279.
[7] Zhang F, Yuan N J, Lian D, et al. Collaborative knowledge base embedding for recommender systems[C]//Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. 2016: 353-362.
[8] Wang Z Y, Yu Q, Wang N, et al. Survey of intelligent question answering research based on knowledge graph[J]. computer engineering and applications, 2020, 56(23): 1-11.
[9] Martinez-Rodriguez J L, Hogan A, Lopez-Arevalo I. Information extraction meets the semantic web: a survey[J]. Semantic Web, 2020, 11(2): 255-335.
[10] Chen B, Zhang Z, Langrené N, et al. Unleashing the potential of prompt engineering in Large Language Models: a comprehensive review[J]. arXiv preprint arXiv:2310.14735, 2023.
[11] Fan W, Ding Y, Ning L, et al. A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models[C]//Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2024: 6491-6501.
[12] Miller G A. WordNet: a lexical database for English[J]. Communications of the ACM, 1995, 38(11): 39-41.
[13] Fellbaum C. WordNet: An electronic lexical database[J]. MIT Press google schola, 1998, 2: 678-686.
[14] Bizer C, Lehmann J, Kobilarov G, et al. Dbpedia-a crystallization point for the web of data[J]. Journal of web semantics, 2009, 7(3): 154-165.
[15] Auer S, Bizer C, Kobilarov G, et al. Dbpedia: A nucleus for a web of open data[C]//international semantic web conference. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007: 722-735.
[16] Subagdja B, Shanthoshigaa D, Wang Z, et al. Machine Learning for Refining Knowledge Graphs: A Survey[J]. ACM Computing Surveys, 2024, 56(6): 1-38.
[17] Zhong L, Wu J, Li Q, et al. A comprehensive survey on automatic knowledge graph construction[J]. ACM Computing Surveys, 2023, 56(4): 1-62.
[18] Schmitt X, Kubler S, Robert J, et al. A replicable comparison study of NER software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate[C]//2019 sixth international conference on social networks analysis, management and security (SNAMS). IEEE, 2019: 338-343.
[19] Che W, Feng Y, Qin L, et al. N-LTP: An open-source neural language technology platform for Chinese[J]. arXiv preprint arXiv:2009.11616, 2020.
[20] Subramanian S, Kawakami K, Dyer C. Neural architectures for named entity recognition[J]. CoRR abs/1603.01360, 2016.
[21] Devlin J. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.
[22] Jia C, Shi Y, Yang Q, et al. Entity enhanced BERT pre-training for Chinese NER[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020: 6384-6396.
[23] Chang Y, Kong L, Jia K, et al. Chinese named entity recognition method based on BERT[C]//2021 IEEE international conference on data science and computer application (ICDSCA). IEEE, 2021: 294-299.
[24] Sun L, Wang J, Zhang K, et al. RpBERT: a text-image relation propagation-based BERT model for multimodal NER[C]//Proceedings of the AAAI conference on artificial intelligence. 2021, 35(15): 13860-13868.
[25] Souza F, Nogueira R, Lotufo R. Portuguese named entity recognition using BERT-CRF[J]. arXiv preprint arXiv:1909.10649, 2019.
[26] Hu S, Zhang H, Hu X, et al. Chinese Named Entity Recognition based on BERT-CRF Model[C]//2022 IEEE/ACIS 22nd International Conference on Computer and Information Science (ICIS). IEEE, 2022: 105-108.
[27] Dai Z, Wang X, Ni P, et al. Named entity recognition using BERT BiLSTM CRF for Chinese electronic health records[C]//2019 12th international congress on image and signal processing, biomedical engineering and informatics (cisp-bmei). IEEE, 2019: 1-5.
[28] Pawar S, Palshikar G K, Bhattacharyya P. Relation extraction: A survey[J]. arXiv preprint arXiv:1712.05191, 2017.
[29] Liu C Y, Sun W B, Chao W H, et al. Convolution neural network for relation extraction[C]//International conference on advanced data mining and applications. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013: 231-242.

[30]Smirnova A, Cudr é -Mauroux P. Relation extraction using distant supervision: A survey[J]. ACM Computing Surveys (CSUR), 2018, 51(5): 1-35.
[31] Lin Y, Shen S, Liu Z, et al. Neural relation extraction with selective attention over instances[C]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2016: 2124-2133.
[32] Peng H, Gao T, Han X, et al. Learning from context or names? an empirical study on neural relation extraction[J]. arXiv preprint arXiv:2010.01923, 2020.
[33] Alt C, Hübner M, Hennig L. Fine-tuning pre-trained transformer language models to distantly supervised relation extraction[J]. arXiv preprint arXiv:1906.08646, 2019.
[34] Han X, Gao T, Yao Y, et al. OpenNRE: An open and extensible toolkit for neural relation extraction[J]. arXiv preprint arXiv:1909.13078, 2019.
[35] Bizer C, Heath T, Berners-Lee T. Linked data-the story so far[M]//Linking the World's Information: Essays on Tim Berners-Lee's Invention of the World Wide Web. 2023: 115-143.
[36] Nasar Z, Jaffry S W, Malik M K. Named entity recognition and relation extraction: State-of-the-art[J]. ACM Computing Surveys (CSUR), 2021, 54(1): 1-39.
[37] Brown, T., et al. "Language models are few-shot learners." NeurIPS (2020).
[38] Touvron, H., et al. "LLaMA: Open and efficient foundation language models." arXiv:2302.13971 (2023).
[39] Mosbach, M., et al. "On the stability of fine-tuning BERT: Misconceptions, explanations, and strong baselines." ICLR (2021).
[40] Hu, E., et al. "LoRA: Low-rank adaptation of large language models." ICLR (2021).
[41] Liang, Y., et al. "GPT4Tool: Connecting large language models with massive tools via instruction tuning." ACL (2023).
[42] Houlsby, N., et al. "Parameter-efficient transfer learning for NLP." ICML (2019).
[43] Li, X., et al. "Prefix-tuning: Optimizing continuous prompts for generation." ACL (2021).
[44] Lester, B., et al. "The power of scale for parameter-efficient prompt tuning." EMNLP (2021).
[45] Zaken, E., et al. "BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models." AACL (2022).
[46] Dettmers, T., et al. "QLoRA: Efficient finetuning of quantized LLMs." NeurIPS (2023).
[47] Zhang, Y., et al. "Knowledge graph completion for high-speed railway turnout maintenance based on multi-level KBGC model." MDPI Actuators (2024).
[48] Chen, L., et al. "Exploring bridge maintenance knowledge graph by leveraging graph data mining." Automation in Construction (2024).
[49] Wang, Q., et al. "A knowledge graph-based approach for exploring railway operational accidents." Reliability Engineering & System Safety (2021).
[50] Liu, W., et al. "Knowledge graph construction based on joint model for equipment maintenance." MDPI Mathematics (2023).
[51] Zhou, H., et al. "Multi-domain fusion for cargo UAV fault diagnosis knowledge graph." Journal of Intelligent Manufacturing (2024).
[52] Cabot, P., et al. "REBEL: Relation extraction by end-to-end language generation." EMNLP (2021).
[53] Hsieh, C., et al. "In-context learning for few-shot knowledge graph completion." AAAI (2024).
[54] Wang, Y., et al. "ClinicalKG: Automatic construction of clinical knowledge graphs using LLMs." JAMIA (2023).
[55] Zhou W, Zhang S, Gu Y, et al. "Universalner: Targeted distillation from large language models for open named entity recognition." arXiv preprint arXiv:2308.03279, 2023.
[56] Guo Q, Dong Y, Tian L, et al. "BANER: Boundary-aware LLMs for few-shot named entity recognition". arXiv preprint arXiv:2412.02228, 2024.
[57] Wang J, Zhang L, Lee W S, et al. "When phrases meet probabilities: enabling open relation extraction with cooperating large language models" Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2024.