applied sciences

MDPI

Article

Ontology-Constrained Knowledge Graph Construction and Relation-Grounded Question Answering for Specialty Crop Knowledge Services: A Jujube Industry Case Study

Jian Wang †, Jianzhong Feng †, Dan Wang * and Yuanyuan Tu *

Institute of Agricultural Information, Chinese Academy of Agricultural Sciences, Beijing 100081, China; wangjian02@caas.cn (J.W.); fengjianzhong@caas.cn (J.F.)

* Correspondence: wangdan01@caas.cn (D.W.); tuyuanyuan@caas.cn (Y.T.)

† These authors contributed equally to this work.

Featured Application

The ontology-constrained knowledge-engineering framework is intended for specialty crop settings in which domain knowledge is fragmented, annotated corpora are limited, and structured relation-grounded retrieval is required. The jujube case illustrates a reproducible workflow for lifecycle-oriented ontology design, curated knowledge graph construction, ontology-constrained relation extraction, and graph-verifiable retrieval. Adaptation to other crops would require domain-specific ontology revision, rule adjustment, and expert validation.

Abstract

Specialty crop industries generate heterogeneous knowledge across cultivation, pest and disease management, post-harvest handling, quality control, and marketing, yet this knowledge is often dispersed across technical documents, web resources, relational datasets, and expert experience. Taking the jujube (*Ziziphus jujuba* Mill.) industry as a case study, this study develops a reproducible ontology-constrained knowledge-engineering framework. A lifecycle-oriented domain ontology and a curated knowledge graph prototype were constructed from verified ontology-aligned records, while unstructured texts were used to evaluate prospective information-extraction components. Four BERT-based named entity recognition models were compared under leakage-controlled five-fold cross-validation, and versioned ontology-constrained trigger rules were evaluated under a gold-entity setting. The final ontology contained 457 RDF triples, 38 classes, 13 object properties, and 71 data properties, and logical consistency checks detected no inconsistent named classes. BERT-Softmax obtained the highest mean NER F1-score (0.328 ± 0.146) in this small corpus, but the fold variability does not support a general claim of superiority over CRF-based architectures. Under gold entities, the refined trigger rules achieved an F1-score of 0.683 ± 0.142; in a stricter diagnostic using automatically predicted fine-grained entities, end-to-end relation extraction declined to 0.035 ± 0.054. On a 93-question benchmark generated from verified graph triples, the rule-graph QA engine achieved a token-level F1-score of 0.973 and a tail-set exact accuracy of 98.9%, primarily reflecting relation mapping and graph retrieval correctness. A separate 20-question human-authored challenge audit found three partially supported questions and no fully supported question, illustrating the boundary beyond graph-grounded retrieval. The results provide case-specific, exploratory evidence that explicit ontology constraints can support reproducible knowledge organization and verifiable retrieval under limited domain data; they do not establish deployment-level or unrestricted agricultural QA performance.

Check for updates

Academic Editor: Jose María Alvarez Rodríguez

Received: 8 July 2026
Revised: 2 August 2026
Accepted: 4 August 2026
Published: 5 August 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

Keywords: specialty crop knowledge services; jujube industry; domain ontology; knowledge graph; relation extraction; relation-grounded question answering

1. Introduction

Specialty crop industries are increasingly dependent on timely, accurate, and scenario-oriented knowledge services. Unlike major staple crops, specialty crops often involve highly localized varieties, diverse cultivation practices, region-specific pest and disease patterns, differentiated post-harvest handling methods, and market-oriented quality requirements. These characteristics generate large amounts of heterogeneous knowledge across the production and service chain, including agronomic manuals, technical standards, expert experience, relational databases, web resources, and unstructured textual documents. However, such knowledge is often scattered across disconnected sources and expressed in inconsistent formats, making it difficult for growers, extension workers, researchers, and agricultural service providers to retrieve, integrate, and reuse domain knowledge efficiently.

Agricultural knowledge services aim to transform fragmented agricultural information into usable knowledge that can support decision-making, technical consultation, production management, and intelligent service delivery. In practical agricultural scenarios, users usually do not need isolated documents or keyword-matched text fragments; rather, they require structured answers that connect varieties, cultivation operations, environmental conditions, pests and diseases, control measures, quality indicators, and market-oriented service information. Therefore, the central challenge is not only data collection, but also semantic organization: agricultural knowledge must be represented in a form that is machine-readable, logically connected, and adaptable to different service tasks [1,2].

Ontology and knowledge graph technologies provide an effective pathway for semantic organization in agricultural knowledge services. A domain ontology defines the core concepts, hierarchical classes, properties, and semantic constraints within a specific field, thereby providing a formal schema for knowledge representation. Based on such a schema, knowledge graphs can further integrate structured, semi-structured, and unstructured resources into entity–relation networks, supporting semantic retrieval, relation reasoning, knowledge discovery, and question answering. In agriculture, ontology- and knowledge-graph-based approaches have been applied to crop management, pest and disease diagnoses, agricultural recommendation, smart extension services, and domain-specific decision support. These studies demonstrate the potential of semantic technologies to bridge the gap between heterogeneous agricultural resources and user-oriented intelligent services [3–16].

Despite this progress, several limitations remain in existing agricultural knowledge graph studies. First, many studies focus on general agricultural concepts or single technical tasks, while the full lifecycle knowledge structure of specialty crop industries has received less systematic attention. Second, information extraction from small, domain-specific agricultural corpora remains challenging. Deep learning models such as BERT-based sequence labeling architectures can support named entity recognition, but their performance may be unstable when annotated corpora are limited and entity types are sparse. Third, relation extraction in agricultural domains often requires explicit use of domain knowledge because many relations are governed by stable agronomic patterns and ontology constraints rather than purely statistical text features. Fourth, question answering systems are frequently demonstrated through examples, but they are less often evaluated through graph-verifiable

benchmarks that can distinguish structured relation retrieval from open-ended natural language generation [17–26].

The jujube (*Ziziphus jujuba* Mill.) industry provides a representative case for studying these issues. Jujube is an economically important specialty crop in arid and semi-arid regions, with knowledge requirements covering variety selection, orchard establishment, seedling propagation, field management, pruning, pest and disease control, harvesting, grading, storage, packaging, transportation, and marketing. These knowledge elements are closely related but are often dispersed across technical books, web portals, expert documents, structured datasets, and local production experience. The resulting fragmentation limits the efficiency of knowledge sharing and makes it difficult to provide precise and personalized knowledge services for growers and agricultural technicians. A lifecycle-oriented ontology and knowledge graph can help organize these heterogeneous resources and support structured retrieval and intelligent question answering in the jujube industry.

Taking the jujube industry as a case study, this study develops an ontology-constrained knowledge graph framework for specialty crop knowledge services. The framework integrates lifecycle-oriented ontology construction, curated knowledge graph development, named entity recognition, ontology-constrained relation extraction, and relation-grounded question answering. Rather than treating a single model architecture as the core innovation, the study emphasizes reproducible knowledge engineering and evaluates the contribution of different components through named entity recognition experiments, relation extraction ablation analysis, and graph-verifiable question answering evaluation.

The main contributions of this study are threefold. First, a lifecycle-oriented jujube industry ontology and curated knowledge graph prototype were constructed from verified ontology-aligned resources, and the final ontology was subjected to reproducible structural and logical consistency checks. Second, an ontology-constrained information extraction workflow was evaluated through a descriptive comparison of four BERT-based NER variants, versioned and frozen trigger-rule ablations under gold entities, and a strict end-to-end diagnostic using automatically predicted fine-grained entities. Third, a 93-question relation-grounded benchmark was constructed to evaluate graph-verifiable retrieval, while a separate pre-existing set of 20 human-authored questions was retained to examine coverage beyond graph-derived questions. Together, these components evaluate both the utility and the boundary of the proposed knowledge-engineering framework without treating a single model architecture as the central innovation.

The remainder of this paper describes the construction and evaluation of the proposed framework, including data preparation, ontology design, knowledge graph construction, information extraction, and relation-grounded question answering. It then discusses the experimental findings, practical implications, limitations, and potential transferability of the framework to other specialty crop knowledge service scenarios.

2. Materials and Methods

2.1. Overall Framework

This study developed an ontology-constrained knowledge graph framework for specialty crop knowledge services, using the jujube industry as a representative case. The framework was designed to organize heterogeneous domain knowledge, support interpretable information extraction, and enable graph-verifiable question answering. Rather than relying on a single end-to-end model, the framework integrates ontology engineering, knowledge graph construction, named entity recognition, ontology-constrained relation extraction, and relation-grounded question answering into a reproducible workflow.

The overall workflow is shown in Figure 1, which links data preparation, ontology design, graph construction, information extraction, relation-grounded question answering, and evaluation into a reproducible knowledge-engineering pipeline.

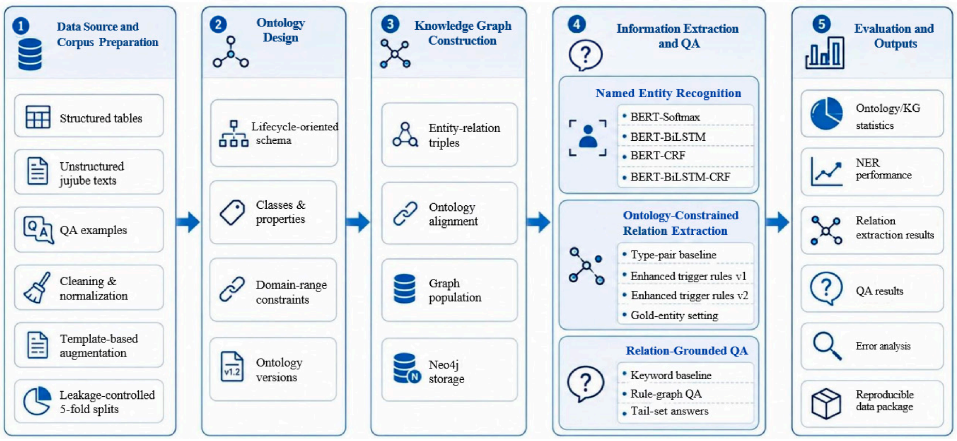

Figure 1. Overall workflow of the ontology-constrained knowledge graph and relation-grounded QA framework. The workflow integrates data preparation, lifecycle-oriented ontology design, knowledge graph construction, information extraction, relation-grounded QA, and evaluation outputs. The figure highlights how ontology constraints connect graph construction, relation extraction, and graph-verifiable retrieval in a reproducible knowledge-engineering pipeline.

The overall workflow consists of six major stages. First, multi-source jujube industry resources were collected and organized, including structured tables, semi-structured web and database records, technical documents, and annotated textual samples. These resources were used to prepare a curated domain corpus and a knowledge graph prototype. Second, a lifecycle-oriented jujube industry ontology was constructed using a top-down strategy. The ontology defined the major concepts, class hierarchies, object properties, data properties, and semantic constraints required to represent knowledge across the jujube production and service chain. Third, structured and semi-structured knowledge resources were mapped into a graph-compatible format, whereas unstructured textual resources were retained to evaluate prospective named entity recognition and relation extraction components rather than directly populate the final curated graph [3–16].

Fourth, named entity recognition was treated as a prospective graph-population component. Four BERT-based sequence-labeling variants were compared under five-fold cross-validation: BERT-Softmax, BERT-BiLSTM, BERT-CRF, and BERT-BiLSTM-CRF. This design allowed the study to evaluate whether more complex sequence-labeling structures improved entity extraction performance under a limited, domain-specific agricultural corpus. Fifth, relation extraction was conducted under ontology and entity-type constraints. A simple type-pair rule baseline was first established, and enhanced trigger-based rules were then introduced to capture stable agronomic relation patterns, such as variety–trait, disease–symptom, technique–material, and parameter–value relations. This setting was evaluated as gold-entity relation extraction, meaning that the relation extraction module was assessed using manually verified entity spans rather than end-to-end predicted entities. A separate strict end-to-end diagnostic supplied automatically predicted fine-grained entities to the frozen v2 rules to quantify error propagation through the complete extraction chain. In the present prototype, NER was evaluated as a prospective graph-population component rather than as a direct source of the final curated knowledge graph. The curated graph and the relation-grounded QA benchmark were constructed from verified ontology-aligned records and triples, while the NER experiment was used to assess the feasibility and limitations of automatic entity identification under limited annotated data [17–22].

Finally, a relation-grounded question answering benchmark was constructed to evaluate graph-verifiable knowledge retrieval. Each benchmark question was linked to a head entity and relation type in the knowledge graph, and the corresponding tail entity set was used as the gold answer. A rule-graph QA engine was compared with a keyword sentence baseline using token-level precision, recall, F1-score, and tail-set exact accuracy. This evaluation focused on structured relation retrieval rather than open-ended natural language generation, thereby making the QA results reproducible and directly verifiable against the graph [23–26]. Because the questions and gold answers were derived from verified graph triples, the benchmark primarily evaluates relation mapping and graph retrieval implementation within the prototype. In addition, 20 pre-existing human-authored questions were audited separately as a small challenge set for examining graph coverage beyond the relation-grounded benchmark.

The framework was evaluated from four perspectives: ontology and knowledge graph statistics, named entity recognition performance, ontology-constrained relation extraction performance, and relation-grounded question answering performance. The complete workflow was designed to be reproducible through shared data files, experimental outputs, and notebook-based evaluation scripts.

2.2. Data Sources and Corpus Preparation

The data used in this study were organized as a curated domain dataset for jujube industry knowledge services. The dataset covered multiple knowledge forms, including ontology files, structured knowledge graph tables, annotated NER/RE samples, and question answering examples. The purpose of data preparation was not to construct a large-scale agricultural corpus, but to establish a controlled and verifiable knowledge-engineering dataset suitable for ontology design, graph construction, information extraction evaluation, and relation-grounded QA benchmarking.

Table 1 summarizes the main data components, their preparation methods, scale, and experimental uses.

Selected initial domain assets, including the ontology-aligned relation inventory and the 20 human-authored test questions, originated from the project "Research on Integrated Demonstration and Application Technologies for 'Internet Plus' Smart Agriculture in Kunyu City" (Project No. 2019AB002). The original knowledge resources were collected from technical materials, structured and semi-structured jujube industry records, web-based agricultural resources, and domain knowledge organized during ontology construction. These resources covered major stages and service needs of the jujube industry, including variety information, orchard establishment, seedling propagation, field management, pruning, soil and water management, pest and disease control, harvesting and grading, storage, packaging, transportation, sales, and quality-related attributes. The knowledge graph data were stored in multiple CSV tables corresponding to different knowledge groups. The ontology files were maintained in OWL format and used as the semantic schema for class definition, property constraints, and graph construction.

For unstructured knowledge extraction, an annotated NER/RE corpus was prepared from selected jujube domain sentences. The original annotated corpus contained 30 sentence-level records, 156 entity mentions, and 117 initially annotated relation instances. One annotator completed the initial annotation of the 30 sentence-level records following the project annotation schema. One coauthor and one master's student subsequently reviewed the entity spans, fine-grained entity types, relation labels, and relation endpoints. Ambiguous cases were resolved through discussion with reference to the annotation schema and ontology definitions. This was an initial-annotation-plus-review workflow rather than independent double annotation; consequently, no inter-annotator agreement statistic was

computed. During data cleaning, entity boundary offsets were checked and corrected, entity labels were standardized, and relation endpoints were verified against the entity list. After consistency checking, 115 valid relation instances were retained for relation extraction evaluation. The entity labels were further mapped into a controlled set of coarse types to reduce sparsity while preserving fine-grained domain labels for relation extraction. The fine-grained labels were also retained for the strict end-to-end diagnostic.

Table 1. Data sources and corpus statistics.

Component	Source or Preparation Method	Scale	Use in This Study
Ontology files	Top-down ontology design and iterative revision	3 OWL versions	Ontology evolution and final semantic schema
Final ontology	Jujube ontology v3	457 RDF triples; 38 classes; 13 object properties; 71 data properties	Semantic schema for graph construction
KG tables	Structured and semi-structured jujube industry records	18 tables; 134 records; 405 non-null attribute values	Curated graph prototype
Original NER/RE corpus	Manually cleaned sentence-level records	30 sentences; 156 entity mentions; 115 valid relations	Gold samples and leakage-controlled evaluation
Augmented NER/RE corpus	Relation-preserving template augmentation	228 augmented sentences	Training support and relation pattern coverage
Final NER/RE corpus	Original + augmented samples	258 sentences; 612 entity mentions; 343 relation instances	NER and relation extraction experiments
Original QA examples	Open-ended jujube knowledge questions	20 questions	Challenge set and limitation analysis
Relation-grounded QA benchmark	Generated from verified graph triples	93 graph-verifiable questions	Main QA evaluation

Because the original corpus was small, relation-preserving template augmentation was used to improve experimental coverage. Template augmentation was based on verified entity–relation triples and predefined linguistic patterns. For example, disease–symptom, technique–material, variety–trait, and parameter–value relations were transformed into English sentences while preserving the original semantic relation. This process produced 228 augmented sentence records. Together with the 30 cleaned original records, the final NER/RE corpus contained 258 sentence-level records, 612 entity mentions, and 343 relation instances. All augmented records were explicitly marked as template-derived samples so that they would not be confused with manually collected original text.

To reduce evaluation leakage, the NER experiments used leakage-controlled five-fold splits. Original records and their template-derived descendants were assigned by lineage so that near-duplicate samples from the same source sentence could not appear across training and held-out test data. Each test fold contained approximately six manually cleaned original records, while the training data could include lineage-compatible original and augmented records. The NER task used BIO tagging and was evaluated with precision, recall, F1-score, and token-level accuracy. For relation extraction, evaluation was conducted in a gold-entity setting. This means that the relation extraction rules were evaluated on verified entity pairs, allowing the experiment to isolate the effect of ontology constraints, entity-type constraints, and trigger rules on relation classification. Given the small number

of original test records, the fold means and standard deviations are treated as descriptive, exploratory evidence rather than statistically conclusive model comparisons.

The QA data were prepared in two complementary forms. First, 20 human-authored questions developed during testing of the preceding prototype were retained as a challenge set. These questions predated the relation-grounded benchmark, were not generated from verified graph triples, and are not claimed to represent a sample of farmer-authored consultations. They were used in a competency-style audit of whether the ontology, graph, and retrieval implementation could address the intended knowledge needs. Second, a relation-grounded QA benchmark was generated from verified graph triples. In this benchmark, each question corresponded to a head entity and a relation type, and all matching tail entities were aggregated as the gold answer. After aggregation, 93 graph-verifiable relation-grounded questions were retained for QA evaluation. This design supports automatic verification, while the separate 20-question audit exposes the boundary between graph-derived retrieval and broader human-authored questions.

Overall, the prepared dataset should be understood as a curated knowledge graph prototype and evaluation corpus for a specialty crop knowledge service framework. It was not intended to represent a large-scale open agricultural text corpus. This distinction is important because the experimental objective was to evaluate whether ontology constraints, graph structure, and relation-grounded retrieval can improve the organization and use of fragmented jujube industry knowledge under limited domain data conditions.

2.3. Jujube Industry Ontology Design

The jujube industry ontology was designed as the semantic schema of the proposed knowledge graph. Its purpose was to provide a formal and machine-readable representation of the core concepts, hierarchical structures, properties, and semantic constraints required for organizing lifecycle-oriented jujube industry knowledge. Considering the relatively clear production process and expert-defined knowledge structure of the jujube industry, this study adopted a top-down ontology construction strategy. The ontology was first defined at the conceptual level and then progressively refined into class hierarchies, object properties, data properties, and instance-level constraints [3–5].

The ontology design followed four steps. First, the domain scope was determined according to the production and service chain of the jujube industry. The scope covered planting preparation, seedling propagation, orchard establishment, field cultivation, pruning, soil and water management, pest and disease control, harvesting, grading, storage, packaging, transportation, sales, and quality-related service information. This lifecycle-oriented scope was used to ensure that the ontology was not limited to a single technical stage but could support knowledge organization across production, management, post-harvest logistics, and service-oriented retrieval.

Second, the main classes and hierarchical relationships were defined. The ontology organized domain concepts into a multi-level class structure, including core entities such as jujube varieties, production areas, cultivation techniques, environmental conditions, pests and diseases, control materials, quality attributes, post-harvest operations, storage methods, packaging methods, and market-related information. These classes were arranged according to both agronomic logic and knowledge service requirements. For example, cultivation-related classes were connected to orchard establishment, pruning, soil management, and water management, while post-harvest classes were connected to grading, storage, packaging, transportation, and sales. The lifecycle-oriented structure and the main ontology modules are summarized in Figure 2, which shows how production, management, post-harvest, quality, and service-related knowledge are organized under a unified semantic schema.

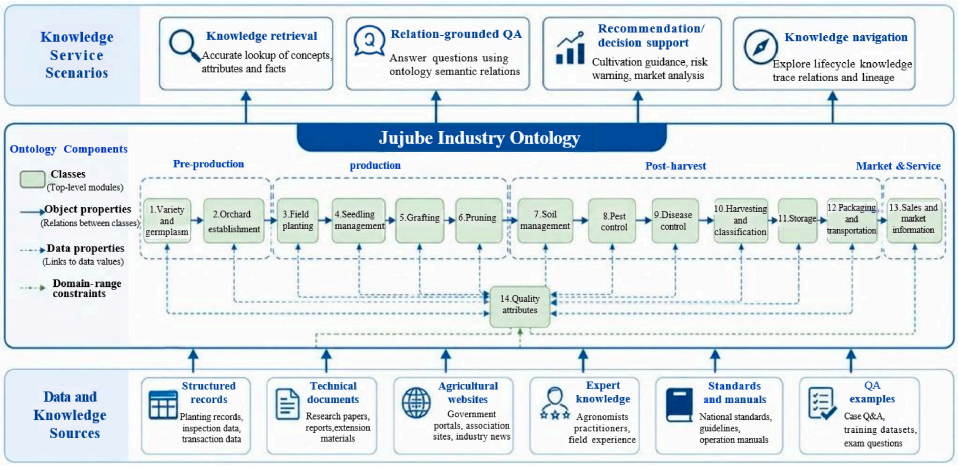

Figure 2. Lifecycle-oriented ontology schema of the jujube industry. The ontology organizes jujube industry knowledge across pre-production, production, post-harvest, market, and service-oriented modules. Classes, object properties, data properties, and domain–range constraints jointly provide the semantic schema for knowledge graph construction, relation extraction, and relation-grounded QA.

Third, object properties were defined to represent semantic relations between classes. These properties described relations such as variety–origin, variety–trait, disease–symptom, disease–pathogen, pest–damage, technique–material, technique–operation stage, product–quality attribute, and standard–requirement. Object properties were designed not only for graph representation, but also for relation extraction and question answering. Therefore, relation definitions were required to be explicit enough to constrain candidate entity pairs and stable enough to support rule-based matching.

Fourth, data properties were defined to describe literal attributes of entities, such as names, aliases, descriptions, growth conditions, environmental parameters, quality indicators, dosage values, storage temperatures, packaging specifications, and market-related attributes. These data properties allowed structured values and textual descriptions to be linked to ontology classes. Together with object properties, they provided the semantic constraints required for graph construction, information extraction, and relation-grounded QA.

The ontology was implemented in OWL format to support formal representation and interoperability. Classes were represented as owl: Class, hierarchical dependencies were expressed through subclass relations, object properties were represented as owl:ObjectProperty, and data properties were represented as owl:DatatypeProperty. Domain and range constraints were assigned to properties where applicable, enabling the ontology to restrict relation types between entity classes. The final version of the ontology was used as the semantic backbone for knowledge graph construction and as the constraint layer for relation extraction and graph-based question answering [6,7].

To improve consistency, ontology construction was conducted iteratively. Candidate classes and properties were first derived from domain knowledge and existing structured resources, and then revised according to entity types, relation patterns, and graph construction requirements observed during corpus preparation. Ambiguous or overlapping classes were merged where necessary, while relation types that were too broad for graph retrieval were refined into more operational forms. This iterative process ensured that the ontology could serve both as a conceptual representation of the jujube industry and as a practical schema for downstream computational tasks [3–5].

After finalization, the ontology was subjected to machine-checkable structural and logical checks. RDF/OWL parsing was used to verify declared classes and properties, self-subclass axioms, cycles in the named-class hierarchy, and domain–range references. The Pellet reasoner, accessed through Owlready2, was used to identify inconsistent named

classes. The 20 pre-existing human-authored questions were also used in a competency-style coverage audit. These procedures evaluate the internal consistency and retrievability of the represented knowledge; they do not constitute a comprehensive assessment of semantic completeness or a formal multi-expert ontology validation study.

2.4. Knowledge Graph Construction

The jujube industry knowledge graph was constructed from verified structured and semi-structured resources under the constraints of the domain ontology, while unstructured texts were used to evaluate prospective information-extraction components. The goal was to transform fragmented domain knowledge into a graph-based representation that could support semantic retrieval, relation extraction evaluation, and relation-grounded question answering. The graph construction process included data preprocessing, ontology alignment, entity normalization, relation generation, graph storage, and quality checking [8–10].

Structured and semi-structured data were processed through an ontology-guided ETL pipeline. First, raw records were screened, duplicate records were removed, and the remaining records were standardized. Entity names, aliases, units, categorical labels, and attribute fields were normalized to reduce inconsistency across different sources. Second, the cleaned records were organized into relational tables according to knowledge groups such as variety information, orchard establishment, field planting, seedling management, grafting, pruning, soil management, pest control, disease control, harvesting and classification, storage, packaging and transportation, and sales. Third, table fields were aligned with ontology classes and properties. In this step, entity columns were mapped to class instances, relational columns were mapped to object properties, and descriptive or numerical fields were mapped to data properties.

To bridge relational data and semantic graph representation, mapping rules were used to convert structured records into graph-compatible triples. The mapping process followed the ontology schema and produced subject–predicate–object structures that could be imported into a graph database. For example, a variety record could generate triples linking the variety to its origin, traits, suitable production areas, quality indicators, and management requirements. Similarly, pest and disease records could generate triples linking symptoms, causal agents, affected parts, control materials, and recommended techniques [6–10].

Unstructured textual resources were used to evaluate the information extraction workflow described in the following sections. Named entity recognition was assessed as a potential graph-population component, while ontology-constrained relation extraction was evaluated using verified entity pairs. Candidate triples generated from textual resources were normalized against ontology classes and relation types, but the final curated graph and QA benchmark relied on verified ontology-aligned triples to ensure evaluation consistency. The revised graph construction and information extraction pipeline is summarized in Figure 3.

The graph was stored in Neo4j, where entities were represented as nodes and semantic relations were represented as edges. Node labels corresponded to ontology classes or knowledge groups, and edge types corresponded to object properties or normalized relation types. Attribute values were stored as node properties where appropriate. This representation enabled both graph traversal and relation-grounded retrieval. For example, a query about a jujube variety could retrieve its origin, traits, cultivation requirements, associated diseases, post-harvest handling methods, and quality attributes through connected graph paths [8–10].

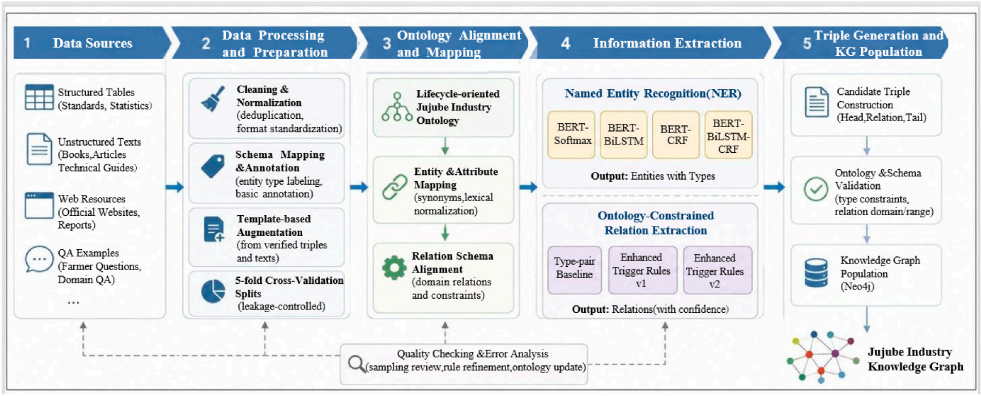

Figure 3. Knowledge graph construction pipeline from heterogeneous resources to graph-based knowledge representation. Structured tables, unstructured texts, web resources, and QA examples are processed through cleaning, schema mapping, ontology alignment, information extraction, triple generation, and graph population. Quality checking and error analysis are used as feedback mechanisms for corpus refinement, rule adjustment, and ontology updating. Only verified ontology-aligned records and triples were retained in the final curated graph; NER and automatic relation extraction were evaluated as prospective graph-population components.

Quality checking was conducted at three levels. At the schema level, class and property definitions were checked for naming consistency, hierarchical clarity, and domain–range compatibility. At the data level, duplicated records, empty fields, inconsistent labels, and malformed values were identified and revised. At the relation level, extracted triples were checked against entity lists, ontology-defined relation types, and relation endpoint constraints. Relations with missing head or tail entities were excluded from formal evaluation, while uncertain candidates were retained only when they could be verified by ontology constraints or domain patterns.

The constructed graph should be understood as a curated knowledge graph prototype rather than a large-scale open agricultural graph. Its value lies in providing a controlled, interpretable, and reproducible graph structure for evaluating ontology-constrained extraction and relation-grounded QA. The final graph dataset consisted of multiple knowledge tables covering major jujube industry processes, and its detailed statistics are reported in the results section together with ontology statistics.

2.5. Named Entity Recognition Models

Named entity recognition was evaluated as a prospective information extraction component for identifying domain entities from sentence-level jujube industry texts. The purpose of this module was to support knowledge graph population by detecting entity mentions related to varieties, diseases, pests, symptoms, cultivation techniques, materials, environmental conditions, measurement values, product quality attributes, standards, and service-oriented knowledge elements. Its predictions were not used as a direct source of the final curated knowledge graph [17–21].

The NER task was formulated as a sequence-labeling problem. Each sentence in the annotated corpus was converted into BIO tags, where B denotes the beginning of an entity mention, *I* denotes the inside of an entity mention, and O denotes tokens outside any entity. Entity labels were standardized before training to reduce inconsistencies in domain annotation. Both fine-grained domain labels and controlled coarse entity types were retained during data preparation, while the NER experiments focused on the standardized label set used for cross-validation.

The four-model architecture comparison used the standardized coarse label set to reduce sparsity, whereas the strict end-to-end diagnostic used the fine-grained types required by the relation rules.

Four BERT-based sequence-labeling models were compared under the same data splits and evaluation protocol. The first model, BERT-Softmax, used contextual token representations generated by BERT and applied a linear classification layer followed by softmax classification to predict token labels. This model served as a simple but strong contextual baseline. The second model, BERT-BiLSTM, added a bidirectional long short-term memory layer on top of BERT embeddings to capture sequential dependencies before token classification. The third model, BERT-CRF, combined BERT representations with a conditional random field layer to model label transition constraints. The fourth model, BERT-BiLSTM-CRF, integrated BERT, BiLSTM, and CRF components, representing the most complex sequence-labeling architecture among the four variants [17–21].

All models were evaluated using leakage-controlled five-fold cross-validation. The split strategy prevented an original sentence and its template-derived descendants from appearing across training and held-out test data. For each fold, the training set contained lineage-compatible original and augmented records, while the development and test sets were based on manually cleaned original records; each test set contained approximately six original records. This design reduced the risk that models would be evaluated on near-duplicate sentences derived from the same original annotation. It did not, however, remove the uncertainty associated with the small held-out samples.

The same preprocessing and evaluation procedure was used for all four models. Sentences were tokenized with the BERT tokenizer, and BIO labels were aligned with tokenized sequences. Model performance was evaluated using entity-level precision, recall, and F1-score, with token-level accuracy reported as a supplementary indicator. Precision measured the proportion of predicted entities that matched gold annotations, recall measured the proportion of gold entities correctly identified, and F1-score was used as the primary metric because it balances precision and recall. The comparative results are reported in the results section.

Entity-level precision, recall, and F1-score were calculated using the following definitions:

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3}$$

where TP denotes correctly predicted entity mentions, FP denotes predicted entity mentions that do not match the gold annotations, and FN denotes gold entity mentions missed by the model. An entity prediction was counted as correct only when both the entity spans and entity type matched the gold annotation after normalization.

This comparative design examined whether more complex sequence-labeling architectures showed a consistent advantage within the limited jujube corpus. Therefore, no single BERT-based architecture was assumed to be superior a priori. The five-fold means and standard deviations were used for descriptive comparison only; the experiment was not powered for statistical significance testing and should not be interpreted as establishing the general superiority of one decoding structure over another [17–21].

2.6. Ontology-Constrained Relation Extraction

Relation extraction was used to convert verified entity pairs into semantic triples for knowledge graph construction and graph-based retrieval. Unlike open-domain relation extraction, relation extraction in the jujube industry is strongly constrained by domain knowledge. Many relations follow stable agronomic and service-oriented patterns, such as variety–trait, disease–symptom, disease–control material, technique–operation stage, technique–material, parameter–value, standard–requirement, and product–quality relations. Therefore, this study adopted an ontology-constrained rule-based relation extraction strategy rather than relying solely on statistical relation classification [22].

The relation extraction task was evaluated in a gold-entity setting. In this setting, the entity spans and entity types used as inputs were manually verified, and the relation extraction module was evaluated independently from NER errors. This design allowed the experiment to isolate the effect of ontology constraints, entity-type constraints, and trigger rules on relation identification. A predicted relation was considered correct only when the head entity, relation type, and tail entity matched the gold relation after normalization. A separate strict end-to-end diagnostic was conducted to quantify the limitation introduced by automatically predicted entities.

The extraction process consisted of four steps. First, all candidate ordered entity pairs within a sentence were generated. Second, candidate pairs were filtered according to ontology-defined entity-type compatibility. For example, disease-related relations were allowed only between compatible types such as disease, symptoms, pathogens, affected part, chemical material, or control technique. Third, relation trigger patterns were applied to detect explicit or semi-explicit linguistic cues. These included expressions indicating origin, symptom, causation, control, requirement, material use, parameter value, operation stage, quality attribute, comparison, and applicability. Fourth, candidate triples were normalized to the relation inventory and checked against ontology constraints before being retained as predicted relations [22]. The ontology-constrained relation extraction mechanism is summarized in Figure 4.

To evaluate the contribution of ontology and trigger constraints, three relation extraction settings were compared. The first setting was a type-pair rule baseline. It predicted relation types mainly according to observed entity-type pairs in the training data. This baseline tested whether entity-type co-occurrence alone was sufficient for relation extraction. The second setting introduced enhanced trigger rules. These rules combined entity-type constraints with relation-specific lexical and syntactic patterns, allowing the system to identify relations such as disease–symptom, technique–material, parameter–value, standard–requirement, and variety–trait more explicitly. The third setting further refined the enhanced rules by unifying relation-group mapping, expanding domain-specific trigger patterns, and adopting a more conservative fallback strategy when relation evidence was weak.

The enhanced rules were designed to reflect both linguistic cues and agronomic constraints. The ontology-level relation inventory and initial domain patterns originated from the preceding jujube knowledge-service project, whereas the v1 and v2 implementations were explicitly versioned for experimental evaluation. Both rule inventories were frozen before the five-fold evaluation and applied unchanged across all held-out folds. Only the fallback entity-type signature mapping was estimated from the training and development records within each fold; held-out test records were used solely for scoring and post-evaluation error analysis. No rule was added or modified on the basis of errors observed in the corresponding test fold. For example, expressions such as “symptom of”, “caused by”, “controlled by”, “requires”, “contains”, “originates from”, “applied during”, and “used for” were associated with corresponding relation categories when the entity types satisfied

ontology constraints. In addition, domain-specific rules were introduced for jujube-related knowledge patterns, including variety traits, planting density, storage conditions, packaging materials, disease control measures, and quality grading requirements. When multiple candidate relations were possible for the same entity pair, relation selection was determined by the strongest trigger evidence and compatibility with the ontology schema.

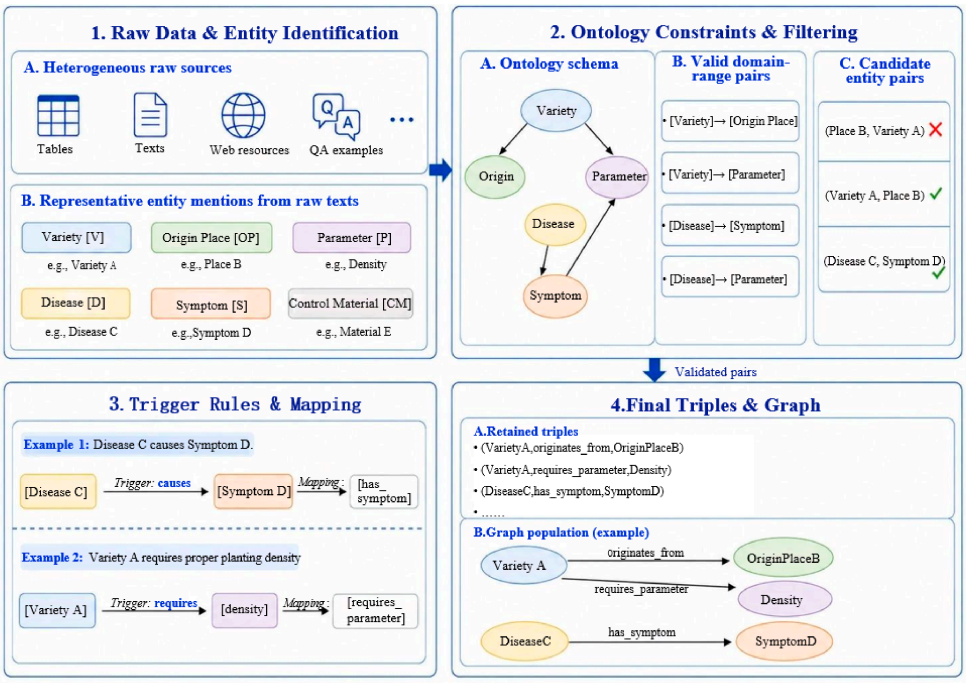

Figure 4. Ontology-constrained relation extraction mechanism based on entity-type constraints and trigger rules. Candidate entity pairs are first generated from raw texts and then filtered using ontology-defined domain–range constraints. Validated pairs are mapped to relation types through trigger rules, producing normalized triples for graph population and relation extraction evaluation. Check marks indicate ontology-compatible candidate pairs, whereas the cross indicates an incompatible candidate pair.

Relation extraction performance was evaluated using precision, recall, and F1-score. Precision was calculated as the proportion of predicted triples that matched gold triples, recall as the proportion of gold triples successfully extracted, and F1-score as the harmonic mean of precision and recall. In addition to fine-grained relation-type evaluation, relation-group-level analysis was used to examine whether the system could identify broader semantic categories even when fine relation labels were difficult to distinguish. Error cases were further reviewed to identify common sources of confusion, such as over-generation of candidate pairs, ambiguity between technique-related relations, and boundary cases involving measurement or quality attributes.

The same precision, recall, and F1-score definitions used for NER were applied to relation extraction, but TP, FP, and FN were calculated at the triple level.

A true positive relation required the head entity, relation type, and tail entity to match the gold triple after normalization. A false positive referred to a predicted triple that did not exist in the gold annotation, while a false negative referred to a gold triple that was not extracted by the rule system. This triple-level evaluation was used to compare the type-pair baseline, enhanced trigger rules, and refined ontology-constrained rules under the same gold-entity setting.

This evaluation design allowed the study to compare a simple type-pair baseline with progressively enhanced ontology-constrained rules. The goal was not to claim a universally optimal relation extraction algorithm, but to examine whether explicit ontol-

ogy constraints and domain trigger patterns can improve relation extraction for a small, specialized agricultural knowledge corpus [22].

A strict end-to-end diagnostic was conducted using the same lineage-controlled folds. BERT-Softmax was trained to predict the fine-grained entity types required by the frozen v2 rule inventory, and its predicted spans and types were supplied directly to the relation module. Entity predictions required an exact span-and-type match, and relation predictions required an exact normalized head–relation–tail match. This diagnostic measured error propagation through the complete extraction chain and was not used to populate the final curated graph.

2.7. Relation-Grounded QA Benchmark

To evaluate whether the constructed knowledge graph could support structured knowledge retrieval, this study developed a relation-grounded question answering benchmark. The benchmark was designed to assess graph-verifiable question answering rather than open-ended natural language generation. Each question was grounded in a verified graph relation and linked to a specific head entity and relation type. The corresponding tail entity or tail entity set was used as the gold answer. This design made the evaluation reproducible because the correctness of an answer could be automatically checked against the graph structure [23–26]. The benchmark therefore assesses graph retrievability, relation-intent mapping, and retrieval implementation correctness within the curated prototype rather than unrestricted natural-language question answering.

The relation-grounded QA benchmark was generated from verified triples in the curated knowledge graph. For each selected relation pattern, a natural language question was created according to the head entity and relation type. For example, a triple in the form of (head entity, relation type, tail entity) could be transformed into a question asking what entity, attribute, material, condition, or process was connected to the head entity through that relation. When multiple tail entities shared the same head entity and relation type, these tails were aggregated into a single gold answer set. This aggregation avoided penalizing the QA system for returning multiple correct graph answers.

The benchmark covered several types of structured agricultural questions, including factual, procedural, causal, and comparative questions. Factual questions asked about entity attributes or direct relations, such as origins, traits, symptoms, materials, or quality indicators. Procedural questions focused on operation-related knowledge, such as cultivation steps, management techniques, storage conditions, or post-harvest processes. Causal questions involved disease, symptoms, pathogens, control, or environmental relations. Comparative questions were used where relation patterns supported comparison between varieties, techniques, or quality attributes. Because the benchmark was derived from verified graph triples, its primary purpose was to test structured graph retrieval rather than unrestricted agricultural consultation.

Two QA engines were compared. The first was a keyword sentence baseline. It retrieved candidate sentences or graph-related text fragments according to keyword overlap between the question and available textual evidence and then returned the most relevant candidate answer. This baseline represented a simple retrieval-oriented strategy that did not explicitly use graph relations. The second was a rule-graph QA engine. It first identified the head entity and relation intent from the question, mapped the relation intent to a normalized graph relation type, and then retrieved the corresponding tail entity set from the graph. The returned answer was therefore directly grounded in the knowledge graph [23–26]. This comparison contrasts simple text retrieval with structured graph retrieval under graph-derived questions and should not be interpreted as a balanced

ranking against contemporary general-purpose QA systems. The benchmark-generation and evaluation workflow is provided in Supplementary Figure S1.

The evaluation used both token-level and graph-level metrics. Token-level precision, recall, and F1-score were calculated by comparing the predicted answer tokens with the gold answer tokens. These metrics measured the textual overlap between predicted and gold answers.

For a predicted answer A_{pred} and a gold answer A_{gold}, token-level precision, recall, and F1-score were defined as follows:

$$P_{token} = \frac{\left|A_{pred} \cap A_{gold}\right|}{\left|A_{pred}\right|} \tag{4}$$

$$R_{token} = \frac{\left|A_{pred} \cap A_{gold}\right|}{\left|A_{gold}\right|} \tag{5}$$

$$F1_{token} = \frac{2 \times P_{token} \times R_{token}}{P_{token} + R_{token}} \tag{6}$$

where A_{pred} and A_{gold} denote the normalized token sets of the predicted answer and the gold answer, respectively. The intersection $A_{pred} \cap A_{gold}$ represents the tokens shared by the predicted and gold answers.

However, token-level metrics may be insufficient for graph-based QA because they can be affected by wording differences, entity order, or the presence of multiple correct tail entities. Therefore, this study also used tail-set exact accuracy as a graph-level metric. Tail-set exact accuracy was calculated as:

$$Acc_{tail} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\left(T_i^{pred} = T_i^{gold}\right) \tag{7}$$

where N denotes the number of relation-grounded questions, T_i^{pred} denotes the predicted tail entity set for the i-th question, T_i^{gold} denotes the corresponding gold tail entity set, and $\mathbf{1}(\cdot)$ is an indicator function that equals 1 when the predicted and gold tail sets are identical after normalization and 0 otherwise.

In addition to answer correctness, average response latency was recorded as a supplementary indicator. The latency comparison was intended to show the computational cost of keyword retrieval and rule-graph retrieval under the experimental setting. However, latency was not treated as the primary evaluation metric because the implementation was a prototype and was not optimized for large-scale deployment. The main focus of the QA benchmark was answer correctness and graph verifiability. No production-scale or controlled scalability inference is drawn from the reported latency values.

In addition to the relation-grounded benchmark, the 20 pre-existing human-authored questions were retained as a separate challenge set. They were not generated from the verified triples and were not treated as representative samples of real farmer consultations. Each question was manually reviewed and categorized as fully supported, partially supported, or unsupported according to whether the graph and retrieval output addressed the intended knowledge need. The challenge-set findings were not combined with the 93-question benchmark metrics; instead, they were used to examine the boundary between structured graph retrieval and broader agricultural questions [23–26].

2.8. Reproducibility and Data Availability

Reproducibility was considered a central requirement in this study. The complete experimental workflow was organized into a versioned package containing ontology files, knowledge graph tables, cleaned and augmented NER/RE corpora, frozen relation-rule inventories, relation-grounded QA data, ontology validation outputs, end-to-end diagnostic outputs, and notebook-based scripts. The purpose of this package was to allow other researchers to inspect the data processing procedure, reproduce the experimental results, and adapt the workflow to other specialty crop knowledge service scenarios.

The revision-stage validation environment used Python 3.12.13, PyTorch 2.11.0+cu128, Transformers 4.41.2, Tokenizers 0.19.1, Hugging Face Hub 0.36.2, seqeval 1.2.2, RDFLib 7.0.0, NetworkX 3.3, and Owlready2 0.46; the NER experiments used the bert-base-cased model. The corresponding environment records and executable notebooks are archived in the Figshare package.

The ontology files were provided in OWL format, including multiple ontology versions used during iterative development. The final ontology version served as the semantic schema for graph construction and relation constraints. The knowledge graph data were provided as structured CSV tables corresponding to major jujube industry knowledge groups. These tables included records related to varieties, orchard establishment, field planting, seedling management, grafting, pruning, soil management, pest control, disease control, harvesting and classification, storage, packaging and transportation, and sales. Together, the ontology and graph tables formed the curated knowledge graph prototype used in this study.

The information extraction data were provided in both cleaned and augmented forms. The cleaned corpus contained manually verified sentence-level records, entity annotations, and relation annotations. The augmented corpus contained relation-preserving template-derived samples, with provenance labels indicating whether a record came from the original cleaned corpus or from template augmentation. The leakage-controlled five-fold splits were also included to make the NER experiments reproducible. For relation extraction, the package included relation type mappings, relation group mappings, rule-based prediction outputs, error summaries, and evaluation tables.

The QA evaluation files included the relation-grounded benchmark dataset, graph triples used as evidence, model predictions, automatic metric summaries, and failed-case review sheets. The relation-grounded benchmark was designed to support automatic evaluation through gold tail entity sets. The original open-ended QA examples were retained separately as a challenge set for qualitative analysis rather than as the main benchmark. This separation allows users of the dataset to distinguish graph-verifiable QA from open-ended agricultural consultation.

The Supplementary Materials provide implementation details without overloading the main text. They document the annotation-and-review workflow, ontology classes and properties, structural and reasoning checks, NER settings and folds, frozen v1/v2 rule inventories, the strict end-to-end diagnostic, relation-grounded benchmark generation, the manually reviewed 20-question challenge audit, selected error cases, and the structure of the reproducibility package. These materials are intended to support methodological transparency and facilitate reuse of the dataset and scripts.

All data and code required to reproduce the reported experiments have been deposited in Figshare at https://doi.org/10.6084/m9.figshare.30945194. The repository contains the curated data package, raw input files, cleaned and audited NER/RE corpora, English template-augmented samples, leakage-controlled five-fold splits, ontology and knowledge graph statistics, BERT-based NER evaluation outputs, frozen v1/v2 rule inventories and relation-extraction outputs, the strict fine-grained BERT-Softmax-to-v2 diagnostic, ontology

structural and Pellet reasoning outputs, the manually reviewed 20-question challenge audit, relation-grounded QA benchmark files, Colab notebooks, software-environment records, a README file, and a SHA-256 manifest. Verified graph data are stored separately from prospective or diagnostic extraction outputs.

3. Results

3.1. Ontology and Knowledge Graph Statistics

The final jujube industry ontology and the curated knowledge graph prototype were first evaluated in terms of structural scale and coverage. The ontology was developed through iterative refinement from earlier versions to the final version used in the experiments. The final ontology contained 457 RDF triples, 38 classes, 13 object properties, and 71 data properties. In addition, it included 30 subclass axioms, 156 domain axioms, and 85 range axioms. These statistics indicate that the ontology provided a compact but explicit semantic schema for organizing jujube industry knowledge and constraining downstream relation extraction and question answering. Table 2 summarizes the main structural statistics of the final ontology and the curated knowledge graph dataset, including ontology-level schema elements and graph-level data records used in the subsequent experiments.

Table 2. Ontology and knowledge graph statistics.

Component	Indicator	Value
Final ontology	RDF triples	457
Final ontology	Classes	38
Final ontology	Object properties	13
Final ontology	Data properties	71
Final ontology	Subclass axioms	30
Final ontology	Domain axioms	156
Final ontology	Range axioms	85
Knowledge graph dataset	Knowledge tables	18
Knowledge graph dataset	Records	134
Knowledge graph dataset	Non-null attribute values	405

Note: Domain and range axioms were counted as RDF domain/range assertion triples. A single property may contribute multiple domain or range assertions when it is associated with multiple ontology classes.

The final OWL file was successfully parsed. No self-subclass axioms or cycles in the named-class hierarchy were detected, and Pellet reported no inconsistent named classes. These results support the internal logical consistency of the represented schema under the current axioms, but they should not be interpreted as evidence of semantic completeness or as a substitute for a dedicated multi-expert ontology validation study. Detailed structural and reasoning checks are summarized in Supplementary Table S15.

The knowledge graph dataset consisted of 18 structured knowledge tables and 134 records, with 405 non-null attribute values. These tables covered major jujube industry knowledge groups, including variety information, orchard establishment, field planting, seedling management, grafting, pruning, soil management, pest control, disease control, harvesting and classification, storage, packaging and transportation, and sales. Although the graph was not intended to be a large-scale open agricultural graph, it provided a controlled and verifiable prototype for testing ontology-constrained knowledge organization, relation extraction, and relation-grounded QA.

The ontology and graph statistics also show that the framework emphasized semantic structure rather than data volume alone. Object properties and domain–range axioms were used to define valid semantic relations between classes, while data properties were used to represent textual, categorical, and numerical attributes of domain entities. This structure allowed the ontology to function both as a conceptual representation of the jujube industry and as a constraint layer for computational tasks. Therefore, the constructed graph should be understood as a curated knowledge-engineering dataset for specialty crop knowledge services.

From the perspective of lifecycle coverage, the graph incorporated knowledge from production preparation to post-harvest and service-oriented retrieval. Production-related knowledge was represented through tables such as orchard establishment, field planting, seedling management, grafting, pruning, and soil management. Protection-related knowledge was represented through pest control and disease control records. Post-harvest and circulation-related knowledge was represented through harvesting and classification, storage, packaging and transportation, and sales records. This coverage supports the use of the graph as a prototype for lifecycle-oriented jujube industry knowledge services.

3.2. NER Performance

The NER experiment compared four BERT-based sequence-labeling models under leakage-controlled five-fold cross-validation. Each held-out test fold contained approximately six manually cleaned original records, so the results are reported as a descriptive, exploratory comparison rather than a statistically conclusive ranking. The purpose of this experiment was to evaluate whether more complex sequence-labeling architectures could improve entity extraction performance under a limited, domain-specific agricultural corpus. The results are summarized in Table 3.

Table 3. Five-fold cross-validation results of BERT-based NER models.

Model	Precision	Recall	F1-Score	Accuracy
BERT-Softmax	0.289 ± 0.131	0.382 ± 0.166	0.328 ± 0.146	0.685 ± 0.072
BERT-BiLSTM	0.342 ± 0.110	0.282 ± 0.099	0.307 ± 0.098	0.556 ± 0.090
BERT-CRF	0.261 ± 0.142	0.360 ± 0.160	0.301 ± 0.152	0.642 ± 0.099
BERT-BiLSTM-CRF	0.343 ± 0.143	0.262 ± 0.099	0.296 ± 0.116	0.567 ± 0.090

Among the four models, BERT-Softmax achieved the highest mean F1-score (0.328 ± 0.146). It also achieved the highest token-level accuracy, with an average accuracy of 0.685 ± 0.072. BERT-BiLSTM-CRF obtained the highest average precision, but its recall was lower than that of BERT-Softmax and BERT-CRF, resulting in a lower F1-score. Within this dataset, the simpler BERT-Softmax configuration provided the highest observed balance between precision and recall, whereas no complex architecture showed a stable overall advantage across the small folds.

The observed ranking is plausibly related to the limited number of original annotated records and the sparse distribution of entity categories. Under such conditions, additional sequence modeling layers may increase model complexity without providing enough training evidence to improve generalization. This interpretation is dataset-specific: the results show only that additional complexity did not yield a consistent benefit in the evaluated jujube folds.

The standard deviations across folds were relatively large, especially for BERT-Softmax and BERT-CRF. This reflects the small size and uneven distribution of the domain-specific test samples. Therefore, the NER results should not be interpreted as evidence that a

simple Softmax classifier is generally superior to CRF-based architectures. No statistical significance claim is made; the results should instead be read as preliminary evidence about model behavior under the current corpus and split design.

These findings support treating NER as a prospective graph-population component rather than the central innovation of the study. The main value of the framework lies not in proposing a new NER architecture, but in integrating entity recognition with ontology-constrained relation extraction and graph-verifiable question answering. The modest and variable performance also justifies retaining manual verification in the current curated graph workflow. The subsequent relation analysis therefore distinguishes gold-entity rule performance from the strict end-to-end diagnostic using automatically predicted fine-grained entities.

3.3. Relation Extraction Performance

The relation extraction experiment evaluated whether ontology constraints and trigger-based rules could improve relation identification under a gold-entity setting. In this evaluation, entity spans and entity types were assumed to be verified, and the task focused on determining whether the correct relation type could be assigned to each entity pair. Three settings were compared: a type-pair rule baseline, enhanced trigger rules, and refined ontology-constrained trigger rules. The v1 and v2 fine-type rule inventories were frozen before five-fold evaluation and were applied unchanged to all held-out folds. Fold-level frozen-rule results are reported in Supplementary Table S16. Table 4 compares the type-pair baseline, enhanced trigger rules v1, and enhanced trigger rules v2 under both coarse and fine entity-type settings.

Table 4. Relation extraction ablation results under coarse and fine entity-type settings.

Method	Entity Type Setting	Precision	Recall	F1-Score
Type-pair rule baseline	Coarse entity types	0.103 ± 0.038	0.288 ± 0.103	0.151 ± 0.054
Type-pair rule baseline	Fine entity types	0.372 ± 0.205	0.350 ± 0.194	0.329 ± 0.144
Enhanced trigger rules v1	Coarse entity types	0.375 ± 0.108	0.822 ± 0.067	0.510 ± 0.115
Enhanced trigger rules v1	Fine entity types	0.500 ± 0.190	0.822 ± 0.067	0.607 ± 0.150
Enhanced trigger rules v2	Coarse entity types	0.444 ± 0.122	0.964 ± 0.038	0.600 ± 0.120
Enhanced trigger rules v2	Fine entity types	0.542 ± 0.168	0.964 ± 0.038	0.683 ± 0.142

Across all three relation extraction settings, fine entity types produced higher F1-scores than coarse entity types, suggesting that fine-grained entity labels provided useful constraints for distinguishing relation types in the jujube domain.

Under the fine entity-type setting, the type-pair rule baseline achieved an average F1-score of 0.329 ± 0.144. This result indicates that relation extraction based only on observed entity-type pairs was insufficient for the jujube domain. Although entity-type compatibility provided a useful constraint, many relation types could not be distinguished by type pairs alone. For example, a technique-related entity and a measurement value may indicate different relations depending on the local linguistic context, such as efficiency, dosage, temperature, spacing, or storage condition.

After trigger rules were introduced, relation extraction performance improved substantially. Under the fine entity-type setting, the enhanced trigger rules v1 achieved an average F1-score of 0.607 ± 0.150, with recall increasing to 0.822 ± 0.067. This improvement shows that relation-specific linguistic cues, when combined with fine-grained entity types, can effectively capture stable agronomic relation patterns. These patterns include disease–

symptom, disease–control material, technique–material, parameter–value, variety–trait, and standard–requirement relations.

The refined ontology-constrained trigger rules v2 further improved performance under the fine entity-type setting, achieving the best average F1-score of 0.683 ± 0.142. Its recall reached 0.964 ± 0.038, indicating that most gold relations could be recovered under the gold-entity setting. The improvement from v1 to v2 was mainly associated with the refinement of relation-group mappings, the addition of domain-specific trigger patterns, and a more explicit distinction among fine-grained relations such as technique efficiency, resource saving, operation stage, parameter requirement, and standard applicability.

However, the precision of the best-performing method remained moderate at 0.542 ± 0.168. This suggests that the enhanced rules were effective in improving coverage but still tended to over-generate candidate triples. The result is consistent with the characteristics of rule-based relation extraction in small domain corpora: explicit ontology constraints and trigger patterns can improve recall, but fine-grained relation boundaries may still be difficult to distinguish when multiple possible relations exist between similar entity types.

Overall, the gold-entity results show that explicit ontology constraints, fine-grained entity types, and trigger patterns can improve relation assignment within this small, specialized corpus. The result does not establish a universally optimal relation extraction algorithm: v2 retained only moderate precision and continued to over-generate candidate triples. Moreover, the gold-entity setting isolates the relation module and should not be interpreted as end-to-end automatic knowledge graph population performance [22].

In the strict end-to-end diagnostic, fine-grained BERT-Softmax achieved precision of 0.197 ± 0.098, recall of 0.249 ± 0.122, F1-score of 0.218 ± 0.106, and token-level accuracy of 0.545 ± 0.071. When these predicted entities were supplied to the frozen v2 rules, end-to-end relation extraction decreased to a precision of 0.022 ± 0.034, recall of 0.086 ± 0.121, and F1-score of 0.035 ± 0.054. The sharp decline indicates that fine-grained entity recovery and candidate-triple over-generation are major bottlenecks in the complete automatic pipeline. The final curated graph therefore continued to rely on verified ontology-aligned records and triples. Fold-level entity and relation results, including endpoint-recovery counts, are provided in Supplementary Table S17.

3.4. Relation-Grounded QA Performance

The relation-grounded QA experiment evaluated whether the constructed knowledge graph could support structured and graph-verifiable knowledge retrieval. Two QA engines were compared: a keyword sentence baseline and a rule-graph QA engine. The keyword baseline retrieved answers according to textual overlap, while the rule-graph QA engine identified the head entity and relation intent from the question and returned the corresponding tail entity set from the graph. Because both the questions and the gold tail sets were derived from verified graph triples, these metrics primarily evaluate the correctness of relation mapping and graph retrieval implementation within the constructed benchmark. They should not be interpreted as independent evidence of general open-ended question answering ability. The comparison contrasts two retrieval mechanisms under controlled graph-derived questions and is not intended as a general QA model benchmark. The evaluation results are presented in Table 5.

Within the relation-grounded benchmark, the rule-graph QA engine obtained higher scores than the keyword sentence baseline. Its token-level F1-score reached 0.973, compared with 0.648 for the keyword baseline. The rule-graph engine also achieved a tail-set exact accuracy of 98.9%, indicating that it was able to retrieve nearly all complete and correct tail entity sets for head–relation queries. These values indicate that the implemented

entity–relation mapping and graph traversal recovered nearly all verified tail sets for the graph-derived questions.

Table 5. Performance comparison of the keyword baseline and rule-graph QA engine on the relation-grounded QA benchmark.

QA Engine	Questions	Token Precision	Token Recall	Token F1-Score	F1 $\geq$ 0.50 Accuracy	Tail-Set Exact Accuracy	Average Latency
Keyword sentence baseline	93	0.579	0.819	0.648	66.7%	82.8%	1.15 ms
Rule-graph QA engine	93	0.966	0.983	0.973	98.9%	98.9%	0.15 ms

The keyword sentence baseline also achieved a relatively high recall of 0.819, but its precision and exact tail-set accuracy were lower. This indicates that keyword retrieval can often locate partially relevant information, but it may return incomplete, redundant, or semantically imprecise answers. In contrast, the rule-graph QA engine directly used normalized graph relations and tail-set aggregation, which reduced irrelevant answer fragments and improved exact matching.

The response latency of the rule-graph QA engine was also lower than that of the keyword baseline in this prototype evaluation. The rule-graph engine required an average of 0.15 ms per query, while the keyword sentence baseline required 1.15 ms. These values are descriptive implementation measurements obtained from a small graph. No production-scale or controlled scalability experiment was conducted, and the latency difference should not be generalized to deployment settings.

The strong QA performance should be understood within the boundary of the benchmark design. The relation-grounded QA benchmark evaluated graph-verifiable structured questions, not unrestricted agricultural consultation. Therefore, the result demonstrates that the constructed graph and relation mapping strategy can support precise retrieval for structured specialty crop knowledge, but it does not imply that the system can fully answer all open-ended questions from growers or extension workers. The separate 20-question challenge audit provides a more demanding view of this boundary.

Figure 5 provides a consolidated visual summary of the principal component-level results reported in Tables 3–5.

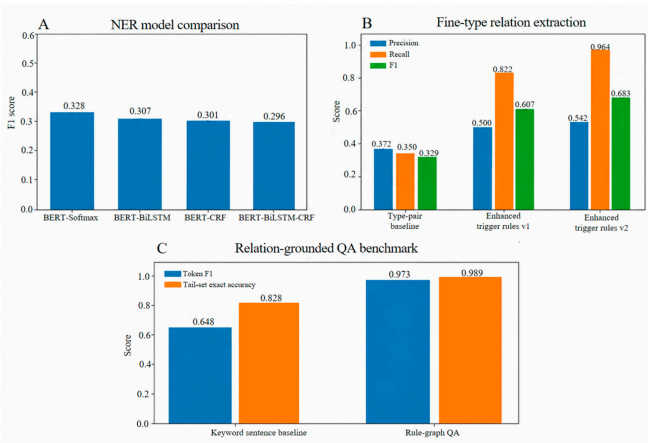

Figure 5. Comparative performance of the evaluated framework components. (**A**) Mean F1-scores of four BERT-based NER models under five-fold cross-validation. (**B**) Mean precision, recall, and F1-scores of the fine-type relation extraction settings. (**C**) Token-level F1-score and tail-set exact accuracy of the keyword sentence baseline and the rule-graph QA engine on the relation-grounded benchmark. For visual clarity, Panels A and B display mean values only; the corresponding standard deviations are reported in Tables 3 and 4.

3.5. Error Patterns in Information Extraction and QA

Error analysis was conducted to examine named entity recognition, gold-entity relation extraction, error propagation in the strict end-to-end pipeline, relation-grounded QA, and the separate 20-question challenge set. A summary of the main error patterns is shown in Table 6.

Table 6. Summary of major error patterns observed in the experiments.

Component	Main Error Pattern	Possible Reason	Implication
NER	Low and unstable F1-scores across folds	Small manually cleaned corpus and sparse entity categories	NER should be treated as a graph population component rather than the core contribution
NER	Lower recall in complex models	Conservative prediction behavior and insufficient training evidence	More model complexity does not guarantee better performance under limited domain data
Relation extraction	Over-generation of candidate triples	Broad trigger coverage and multiple possible relations between similar entity types	Precision can be improved by stricter relation disambiguation
Relation extraction	Confusion among fine-grained technique-related relations	Similar linguistic contexts for efficiency, resource saving, dosage, spacing, and operation stage	More relation-specific examples and domain validation are needed
QA	Failure on some relation-grounded questions	Relation intent mapped to a nearby but incorrect graph relation	Intent mapping and relation normalization require further refinement
Open-ended QA	Poor coverage for broad user-style questions	Many answers require knowledge beyond the curated graph	Open-ended consultation should be treated separately from graph-verifiable QA

For NER, the main issue was the relatively low and unstable F1-score across folds. Although BERT-Softmax achieved the best average F1-score among the four models, the absolute value remained modest. This limitation is mainly related to the small number of manually cleaned sentence records and the uneven distribution of entity categories. Some entity types, such as disease, technique, measurement value, and variety, were easier to identify because they had clearer lexical patterns. In contrast, entities related to symptoms, functions, standards, and context-dependent agronomic conditions were more difficult to recognize consistently.

The NER results also show that increasing model complexity did not necessarily improve performance. BERT-BiLSTM-CRF obtained the highest precision but lower recall, suggesting that it predicted entities more conservatively. Under a small domain corpus, complex sequence-labeling structures may not receive enough training evidence to generalize across sparse entity categories. Therefore, the NER module should be interpreted as a necessary graph population component, not as evidence of a new or superior sequence-labeling model.

For relation extraction, the refined ontology-constrained trigger rules achieved high recall but moderate precision. The most important error pattern was over-generation of candidate triples. Because the enhanced rule system was designed to improve coverage, it sometimes predicted relations for entity pairs that were semantically related but did not match the gold relation. This was especially common for technique-related and

measurement-related relations, where the same sentence could contain multiple nearby entities and several plausible relation types.

Fine-grained relation confusion was another important source of error. Some errors occurred between relations such as technique efficiency, resource saving, operation stage, control method, and material use. For example, a technique and a percentage value may indicate efficiency, reduction rate, survival rate, or resource saving depending on the linguistic context. Similarly, a material associated with a disease may indicate prevention, control, treatment, or chemical application. These cases suggest that rule-based relation extraction benefits from ontology constraints but still requires more relation-specific examples and stricter disambiguation rules.

The strict end-to-end diagnostic exposed a larger error-propagation problem. Fine-grained NER recovered only a limited share of exact entity spans and types, and noisy or fragmented entity predictions created many invalid candidate pairs for the high-recall v2 rules. Consequently, the end-to-end relation F1-score fell to 0.035 ± 0.054 even though the same frozen rules achieved 0.683 ± 0.142 with gold entities. This gap identifies fine-grained entity recovery, rather than relation-label assignment alone, as the main bottleneck for prospective automatic graph population.

For relation-grounded QA, the rule-graph engine performed well on graph-verifiable questions, but the few failed cases reveal the limitation of relation intent mapping. A typical error occurred when the system identified the correct head entity but mapped the question to a nearby relation type. In such cases, the answer was graph-grounded but did not match the intended relation. This indicates that relation-grounded QA depends not only on graph completeness but also on accurate relation intent recognition.

The 20 pre-existing human-authored questions showed a clear boundary of the current prototype. After manual semantic review, three questions were classified as partially supported, 17 as unsupported, and none as fully supported. Partial cases reflected incomplete graph attributes or incomplete answers, whereas unsupported cases commonly required relations, explanations, recommendations, or contextual knowledge not represented in the current graph; one case also retrieved an attribute that did not match the requested property. Because these questions were project-authored rather than farmer-sampled, the audit is a small competency-style challenge analysis rather than a user-performance evaluation. Question-level classifications and primary limitations are reported in Supplementary Table S18.

Overall, the error analysis supports the framework for structured knowledge organization and graph-grounded retrieval within the curated prototype, while also showing substantial limitations in automatic graph population and open-ended agricultural consultation. The low strict end-to-end relation score and the limited coverage of the human-authored challenge set motivate larger independently reviewed corpora, stricter relation disambiguation, broader graph coverage, and continued human verification.

4. Discussion

4.1. Main Findings

The results provide case-specific evidence that ontology-constrained knowledge engineering can organize and retrieve specialty crop knowledge in a controlled, interpretable prototype. The jujube ontology supplied explicit class, property, and domain–range structures, and the structural checks found no self-subclass axioms, named-class cycles, or inconsistent named classes. These checks support internal consistency under the represented axioms, but they do not establish semantic completeness or broad domain validation.

A central finding is that ontology constraints and trigger patterns improved relation assignment when entity spans and fine-grained types were verified. The frozen v2 inventory

reached F1 = 0.683 ± 0.142 under gold entities. Its recall was high, but its moderate precision and false-positive volume show that the rules remain a high-coverage candidate-generation mechanism rather than a deployment-level relation extractor.

The NER comparison showed no stable advantage from adding BiLSTM or CRF components in the current small folds, although BERT-Softmax obtained the highest mean F1-score. This result is descriptive and does not imply general architectural superiority. More importantly, the strict fine-grained end-to-end diagnostic yielded relation F1 = 0.035 ± 0.054, showing that entity-recognition errors propagate sharply into the rule module. Automatic graph population therefore remains a prospective component that requires substantially more data and human verification.

The relation-grounded QA results show that graph-structured retrieval can be highly accurate when questions are generated from verified head–relation patterns and the required tails are present in the graph. The 98.9% exact-tail score mainly reflects graph retrievability and implementation correctness under those conditions. In contrast, the separate human-authored challenge audit produced no fully supported question and only three partially supported questions, confirming that the current graph and rules do not constitute an open-ended agricultural consultation system.

4.2. Implications for Specialty Crop Knowledge Services

The proposed framework has practical implications for specialty crop knowledge services. Specialty crop industries often depend on localized knowledge, diverse varieties, region-specific cultivation practices, pest and disease patterns, post-harvest handling methods, and market-oriented quality requirements. These knowledge elements are usually distributed across technical documents, expert experience, structured records, and web resources. Without semantic organization, users may retrieve isolated text fragments but still fail to obtain structured answers that connect varieties, techniques, conditions, symptoms, materials, standards, and quality attributes [1,2,11–16].

The jujube case shows that a lifecycle-oriented ontology can provide a useful structure for organizing such fragmented knowledge. By connecting production preparation, field management, pest and disease control, post-harvest logistics, quality attributes, and service-oriented retrieval, the ontology helps transform dispersed resources into a unified semantic framework. This structure can support not only knowledge graph construction, but also relation extraction and question answering. In this sense, ontology is not only a conceptual classification tool, but also a computational constraint layer for downstream intelligent services [3–16].

The framework is relevant to settings where annotated data are limited because it makes the ontology, graph, rule inventory, and component-level evidence explicit and inspectable. Nevertheless, the strict end-to-end diagnostic shows that a hybrid design does not remove the need for adequate labeled data: fine-grained entity errors can dominate the downstream relation results. In the current prototype, ontology rules are most defensible as transparent constraints and candidate-generation aids used together with manual verification, rather than as a substitute for robust automatic extraction [17–22].

Nevertheless, the implications should not be overstated. The framework is not a universal agricultural intelligence system or a validated farmer-facing service. Its contribution is a reproducible workflow for constructing and auditing a curated specialty crop knowledge graph prototype. Transfer to grapes, walnuts, citrus, tea, medicinal plants, or other crops would require ontology revision, new or adjusted relation rules, expanded annotations, and independent domain validation. The framework is therefore transferable as a knowledge-engineering procedure, not as a ready-made general QA solution.

4.3. Comparison with Existing Agricultural Knowledge Graph Studies

Existing agricultural knowledge graph studies have made important progress in crop knowledge organization, pest and disease diagnosis, agricultural recommendation, intelligent retrieval, and decision support. Many studies emphasize the role of ontologies in standardizing agricultural concepts and the role of knowledge graphs in connecting heterogeneous resources. Compared with these studies, the present work contributes mainly in three aspects [11–16].

First, this study emphasizes lifecycle-oriented knowledge organization for a specialty crop industry. Rather than focusing only on a single task such as pest diagnosis or recommendation, the jujube ontology and graph prototype cover multiple stages of the production and service chain, including varieties, cultivation operations, pest and disease management, quality attributes, post-harvest logistics, and marketing-related information. This broader lifecycle structure is important for specialty crop knowledge services because user questions often cut across different stages of production, management, and circulation.

Second, this study provides a component-level evaluation of the knowledge graph construction workflow. The NER architecture comparison, frozen gold-entity relation-rule ablation, strict end-to-end diagnostic, relation-grounded benchmark, and human-authored challenge audit report both positive and negative evidence. This design identifies where the workflow is reliable and where it fails. In particular, the contrast between gold-entity and end-to-end relation results prevents the rule module from being mistaken for a complete automatic graph-population system [17–26].

Third, this study distinguishes relation-grounded QA from open-ended agricultural question answering. This distinction is important because graph-based QA is most reliable when the question can be mapped to explicit graph relations. Open-ended agricultural consultation often requires broader domain knowledge, contextual judgment, and explanatory reasoning beyond the current graph. By separating the relation-grounded benchmark from the open-ended challenge set, this study avoids overstating QA performance and provides a more transparent evaluation protocol [23–26].

Compared with purely data-driven approaches, the proposed framework places greater emphasis on explicit semantic constraints and interpretability. Compared with purely rule-based systems, it incorporates BERT-based NER and systematic evaluation. The current experiments do not constitute a state-of-the-art comparison against neural relation extraction, retrieval-augmented generation, or large agricultural language models. With only 30 original annotated sentences, 115 valid relations, and a graph-derived QA benchmark, broader model comparisons would not provide a reliable like-for-like ranking; such comparisons require larger independently annotated corpora and external question sets [17–26].

4.4. Limitations

This study has several limitations. First, the dataset is a curated knowledge-engineering prototype rather than a large-scale open agricultural corpus. The 30 original NER/RE records were produced through one-person initial annotation followed by two-person content review, not independent double annotation, so no inter-annotator agreement statistic is available. The manually cleaned NER/RE corpus was small, and template augmentation was used to improve relation pattern coverage. Although lineage-controlled splits reduced direct template leakage, each held-out fold contained only about six original records. The NER and gold-entity relation results are therefore exploratory and descriptive; no statistical significance or general model-superiority claim is made.

Second, the relation extraction strategy depends on ontology constraints and domain-specific trigger rules. This design improves interpretability and performs well under limited

data conditions, but its portability is not automatic. The best gold-entity precision remained moderate, and the strict end-to-end relation F1-score was only 0.035 ± 0.054. This gap shows that the current evidence supports the rule module under verified entities, not reliable automatic knowledge graph population. Porting the approach to another crop would require revised entity types, triggers, ontology constraints, and independent validation.

Third, the QA evaluation focused on relation-grounded and graph-verifiable questions. The 93 questions and their gold tail sets were generated from verified graph triples, so the high scores measure graph retrievability, relation mapping, and implementation correctness rather than independent generalization to unrestricted questions. The separate 20-question human-authored challenge audit found three partially supported and 17 unsupported questions, with no fully supported question. Although this small set predates the graph-derived benchmark, it was project-authored rather than sampled from farmers and should be interpreted only as a competency-style boundary analysis.

Fourth, the ontology and graph remain limited in scale, coverage, and external validation. The Pellet check found no inconsistent named classes under the current axioms, but this does not demonstrate semantic completeness, comprehensive expert agreement, or correct modeling of every domain relation. Specialty crop knowledge changes over time as new varieties, cultivation techniques, diseases, control materials, standards, and market requirements emerge. The current framework supports structured representation and reproducible evaluation, but it still requires manual or semi-automatic revision for ontology evolution and graph updates. Future work should combine controlled ontology evolution with larger independent annotation and domain-expert review.

Finally, the framework was evaluated in a prototype environment. The reported latency values were obtained from a small graph and are descriptive only; no controlled scaling experiment, user study, platform integration test, or deployment-level robustness assessment was conducted. Production use would require substantially broader data coverage, performance testing, user-centered evaluation, and human-in-the-loop safeguards.

Taken together, these limitations define the scope of the evidence. The study supports a reproducible and inspectable knowledge-engineering workflow for a curated jujube case, but it does not demonstrate a large-scale graph, a robust automatic population pipeline, an unrestricted agricultural QA system, or cross-crop generalization. The negative end-to-end and challenge-set findings are therefore integral to the interpretation of the framework rather than ancillary limitations.

5. Conclusions

This study developed and evaluated an ontology-constrained knowledge-engineering framework for specialty crop knowledge services using a jujube industry case. The framework combines lifecycle-oriented ontology design, a curated graph prototype, exploratory BERT-based NER, versioned ontology-constrained relation rules, and relation-grounded QA. Its contribution is a reproducible component-level workflow rather than a new deep learning architecture, a large-scale agricultural graph, or a deployed consultation system.

The final ontology contained 457 RDF triples, 38 classes, 13 object properties, and 71 data properties, and structural checks detected no self-subclass axioms, named-class cycles, or inconsistent named classes. Together with the 18 curated knowledge tables, the ontology provides an explicit schema and verifiable data basis for the evaluated prototype. These checks support internal consistency but do not establish semantic completeness or broad expert validation.

In the small NER corpus, BERT-Softmax obtained the highest mean F1-score, but the fold variability does not support a general claim that it is superior to CRF-based alternatives. Frozen ontology-constrained rules improved gold-entity relation F1 from 0.329 ± 0.144 for

the type-pair baseline to 0.683 ± 0.142 for v2, while precision remained moderate. When automatically predicted fine-grained entities were used, end-to-end relation F1 declined to 0.035 ± 0.054, showing that the current automatic population chain is not yet reliable.

On the 93-question benchmark generated from verified graph triples, the rule-graph engine achieved token-level F1 = 0.973 and tail-set exact accuracy = 98.9%. These values demonstrate correct graph retrieval after successful relation mapping within the benchmark, not unrestricted open-ended QA capability. The separate 20-question human-authored audit yielded three partially supported and 17 unsupported questions, with none fully supported, further delimiting the current graph coverage.

Overall, the study provides a reproducible, case-specific workflow for organizing, auditing, and retrieving specialty crop knowledge under limited data. Future work should expand independently reviewed corpora, strengthen fine-grained entity recognition and relation disambiguation, broaden graph coverage, conduct formal domain-expert and user evaluations, and test controlled transfer to other crops. Until such evidence is available, the framework should be interpreted as a curated knowledge-engineering prototype with transparent strengths and limitations.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/app16157810/s1: Supplementary Text S1: Data preparation and annotation consistency checking; Table S1: File-level inventory of the experimental dataset and outputs; Table S2: Fine-to-coarse entity type mapping used in NER/RE experiments; Supplementary Text S2: Lifecycle-oriented ontology schema; Table S3: Lifecycle-oriented ontology classes and module descriptions; Table S4: Object properties and domain-range constraints in the jujube industry ontology; Table S5: Representative data properties grouped by ontology module; Supplementary Text S3: NER model settings and evaluation protocol; Table S6: NER model variants and experimental settings; Table S7: Fold-level NER evaluation results; Supplementary Text S4: Ontology-constrained relation extraction rules; Table S8: Relation groups, representative relation types, entity-type constraints, and trigger examples; Table S9: Relation extraction ablation results under coarse and fine entity-type settings; Table S10: Representative relation extraction error cases; Supplementary Text S5: Relation-grounded QA benchmark construction; Figure S1: Relation-grounded QA benchmark and evaluation procedure; Table S11: Relation-grounded QA benchmark construction templates and examples; Table S12: QA benchmark distribution and performance by question type; Table S13: Failed-case review for relation-grounded QA; Supplementary Text S6: Reproducibility package and Figshare file structure; Table S14: Overview of the Figshare reproducibility package structure; Supplementary Text S7: Ontology structural and reasoning validation; Table S15: Summary of ontology structural and Pellet reasoning checks; Supplementary Text S8: Frozen relation-rule evaluation and strict end-to-end diagnostic; Table S16: Fold-level reproducibility check for the frozen relation extraction rules; Table S17: Fine-grained NER and strict end-to-end relation extraction diagnostic; Supplementary Text S9: Human-authored 20-question challenge audit; Table S18: Manual semantic review of the 20 human-authored challenge questions.

Author Contributions: Conceptualization, J.W. and D.W.; methodology, J.F. and Y.T.; software, J.W.; validation, D.W. and Y.T.; formal analysis, J.W.; investigation, Y.T.; resources, D.W.; data curation, J.W.; writing—original draft preparation, J.F.; writing—review and editing, Y.T. and J.W.; visualization, J.W.; supervision, D.W. and Y.T.; project administration, D.W.; funding acquisition, J.W. and D.W. Additionally, J.W. and J.F. contributed equally to this work and share first authorship. D.W. and Y.T. jointly supervised this work. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by the Key Research and Development Project of Henan Province, "Key Technologies and Application Demonstration of Agricultural Intelligent Knowledge Services Based on Big Data and Explainable Artificial Intelligence" (grant no. 241111211100), and by the Basic Scientific Research Funds of the Chinese Academy of Agricultural Sciences, "Research

on Agricultural Scientific Research Big Data Mining and Intelligent Question-Answering Services Based on Generative AI" (Y2026JC09). The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Institutional Review Board Statement: Ethical review and approval were not required for this study.

Informed Consent Statement: Not applicable because this study used curated domain knowledge resources and derived experimental datasets and did not involve identifiable human participants.

Data Availability Statement: The data and experimental outputs supporting the findings of this study are openly available in Figshare at https://doi.org/10.6084/m9.figshare.30945194. The repository contains raw inputs, cleaned and augmented NER/RE corpora, lineage-controlled folds, ontology and graph files, BERT-based NER outputs, frozen v1/v2 rule inventories and relation-extraction results, the strict end-to-end diagnostic, ontology structural and reasoning outputs, relation-grounded QA data, the manually reviewed 20-question challenge audit, Colab notebooks, environment records, a README file, and a SHA-256 manifest. The dataset should be interpreted as a curated knowledge-engineering prototype rather than a large-scale open agricultural corpus or a deployed farmer consultation system.

Acknowledgments: The authors thank Yuan Xue for the initial annotation of the sentence-level corpus and Jun Zou for subsequent annotation review and consistency checking. During the preparation and revision of this manuscript, the authors used ChatGPT (OpenAI, https://chatgpt.com/; model version: GPT-5.6 Sol), Claude (Anthropic, https://claude.ai/; model version: Sonnet 4.6), and Gemini (Google, https://gemini.google.com/; model version: Gemini 3.1 Pro) for English-language polishing, structural refinement, title screening, reference checking, and suggestions on figure presentation and visual improvement. All AI-generated or AI-assisted outputs were critically reviewed and edited by the authors, who take full responsibility for the content of this publication.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Dreyer, S.; Werth, O.; Olivotti, D.; Guhr, N.; Breitner, M.H. Knowledge management systems for smart services: A synthesis of design principles. *E-Serv. J.* **2021**, *13*, 27–67. [CrossRef]
2. Zhao, R.X.; Yang, C.X.; Zheng, J.H.; Li, J.; Wang, J. Agricultural intelligent knowledge service: Overview and future perspectives. *Smart Agric.* **2022**, *4*, 105–125. (In Chinese) [CrossRef]
3. Gruber, T.R. Toward principles for the design of ontologies used for knowledge sharing. *Int. J. Hum.-Comput. Stud.* **1995**, *43*, 907–928. [CrossRef]
4. Uschold, M.; Gruninger, M. Ontologies: Principles, methods and applications. *Knowl. Eng. Rev.* **1996**, *11*, 93–136. [CrossRef]
5. Noy, N.F.; McGuinness, D.L. *Ontology Development 101: A Guide to Creating Your First Ontology*; Stanford Knowledge Systems Laboratory Technical Report KSL-01-05; Stanford Medical Informatics Technical Report SMI-2001-0880; Stanford University: Stanford, CA, USA, 2001.
6. Cyganiak, R.; Wood, D.; Lanthaler, M. RDF 1.1 Concepts and Abstract Syntax. *W3C Recommendation*, 25 February 2014. [CrossRef]
7. W3C OWL Working Group. OWL 2 Web Ontology Language document overview (2nd ed.). *W3C Recommendation*, 11 December 2012.
8. Hogan, A.; Blomqvist, E.; Cochez, M.; D'amato, C.; De Melo, G.; Gutierrez, C.; Kirrane, S.; Gayo, J.E.L.; Navigli, R.; Neumaier, S.; et al. Knowledge graphs. *ACM Comput. Surv.* **2021**, *54*, 71. [CrossRef]
9. Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; Yu, P.S. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 494–514. [CrossRef] [PubMed]
10. Peng, C.; Xia, F.; Naseriparsa, M.; Osborne, F. Knowledge graphs: Opportunities and challenges. *Artif. Intell. Rev.* **2023**, *56*, 13071–13102. [CrossRef] [PubMed]
11. Jonquet, C.; Toulet, A.; Dutta, B.; Emonet, V. Harnessing the power of unified metadata in an ontology repository: The case of AgroPortal. *J. Data Semant.* **2018**, *7*, 191–221. [CrossRef]
12. Baker, T.; Whitehead, B.; Musker, R.; Keizer, J. Global agricultural concept space: Lightweight semantics for pragmatic interoperability. *npj Sci. Food* **2019**, *3*, 16. [CrossRef] [PubMed]

13. Hassani-Pak, K.; Singh, A.; Brandizi, M.; Hearnshaw, J.; Parsons, J.D.; Amberkar, S.; Phillips, A.L.; Doonan, J.H.; Rawlings, C. KnetMiner: A comprehensive approach for supporting evidence-based gene discovery and complex trait analysis across species. *Plant Biotechnol. J.* **2021**, *19*, 1670–1678. [CrossRef] [PubMed]
14. Ngo, Q.H.; Kechadi, T.; Le-Khac, N.A. OAK: Ontology-based knowledge map model for digital agriculture. *arXiv* **2020**, arXiv:2011.11442. [CrossRef]
15. Mu, W.S.; Liu, T.Q.; Miao, Z.W.; Feng, J.Y. Research progress on knowledge graph technology and its application in agriculture. *Trans. Chin. Soc. Agric. Eng.* **2023**, *39*, 1–12. (In Chinese) [CrossRef]
16. Chen, M.; Zhu, J.Z.; Xi, X.T. Knowledge management method of flower diseases and pests based on knowledge graph. *Trans. Chin. Soc. Agric. Mach.* **2023**, *54*, 291–300. (In Chinese)
17. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; Volume 1, pp. 4171–4186. [CrossRef]
18. Lafferty, J.; McCallum, A.; Pereira, F.C.N. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning (ICML 2001)*; Morgan Kaufmann: Burlington, MA, USA, 2001; pp. 282–289. Available online: https://www.cs.columbia.edu/ jebara/6772/papers/crf.pdf (accessed on 3 August 2026).
19. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF models for sequence tagging. *arXiv* **2015**, arXiv:1508.01991.
20. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; Association for Computational Linguistics: San Diego, CA, USA, 2016; pp. 260–270. [CrossRef]
21. Ma, X.; Hovy, E. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2016; Volume 1, pp. 1064–1074. [CrossRef]
22. Zeng, D.; Liu, K.; Lai, S.; Zhou, G.; Zhao, J. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*; Dublin City University and Association for Computational Linguistics: Dublin, Ireland, 2014; pp. 2335–2344. Available online: https://aclanthology.org/C14-1220.pdf (accessed on 3 August 2026).
23. Berant, J.; Chou, A.; Frostig, R.; Liang, P. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2013; pp. 1533–1544. Available online: https://aclanthology.org/D13-1160.pdf (accessed on 3 August 2026).
24. Yih, W.-t.; Chang, M.-W.; He, X.; Gao, J. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2015; Volume 1, pp. 1321–1331. [CrossRef]
25. Lan, Y.; He, G.; Jiang, J.; Jiang, J.; Zhao, W.X.; Wen, J.-R. A survey on complex knowledge base question answering: Methods, challenges and solutions. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI 2021)*; International Joint Conferences on Artificial Intelligence Organization: Darmstadt, Germany, 2021; pp. 4483–4491. [CrossRef] [PubMed]
26. Vollmers, D.; Jalota, R.; Moussallem, D.; Topiwala, H.; Ngomo, A.-C.N.; Usbeck, R. Knowledge graph question answering using graph-pattern isomorphism. In *Further with Knowledge Graphs: Proceedings of the 17th International Conference on Semantic Systems (SEMANTiCS 2021)*; IOS Press: Amsterdam, The Netherlands, 2021; Volume 53, pp. 103–117. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.