Enhancing the Accessibility of Text Data in Decision Making for Capability-based Planning Using Ontology: A Perspective of Semantic Compliance

Mu Lin*, Tongxin Li**, Tao Wang***,
Xiaobo Li **** and Weiping Wang*****

College of Systems Engineering, National University of Defense Technology,
109 Deya Road, Changsha, 410073, Hunan, China
* linmu023@163.com, ** 15545007212@163.com, *** wangtao1976@nudt.edu.cn,
**** lixiaobo@nudt.edu.cn, ***** wangwp@nudt.edu.cn

Mu Lin is a PhD student in the College of Systems Engineering, National University of Defense Technology (NUDT). He currently researches intelligence decision supporting systems. His research interests also include knowledge organization and ontologies.

Tongxin Li is a PhD student in the College of Systems Engineering, National University of Defense Technology (NUDT). Her main research directions include strategic management assessment and knowledge graph representation learning.

Tao Wang is an Associate Professor and master's supervisor of the National University of Defense Technology. He received his Ph.D. in Computer Science and Technology from the College of Computer Science and Technology, National University of Defense Technology (NUDT). He currently conducts research on strategic management assessment and system of systems engineering.

Xiaobo Li is an Associate Professor and master's supervisor of the National University of Defense Technology. He received his Ph.D. in Control Science and Engineering from the College of Systems Engineering, National University of Defense Technology (NUDT). His main research interests include system of systems engineering and simulation.

Weiping Wang is a Full Professor and PhD supervisor at the National University of Defense Technology. He has published more than 100 research papers on multi-agent system modeling and simulation, expert systems, and information management systems. He currently conducts research on intelligence decision supporting systems.

Lin, Mu, Tongxin Li, Tao Wang, Xiaobo Li, and Weiping Wang. 2024. "Enhancing the Accessibility of Text Data in Decision Making for Capability-based Planning Using Ontology: A Perspective of Semantic Compliance". *Knowledge Organization* 51, no. 4: 227-248. 34 references. DOI:10.5771/0943-7444-2024-4-227.

Abstract: Project introduction texts provide decision-makers with essential information for project portfolio planning. These free texts offer valuable information about how developing capabilities align with their strategic goals. Capability-based Planning (CBP) process is concerned with optimizing the project portfolio to realize the planning, engineering, and delivery of these capabilities. However, up to now, the research on enhancing accessibility to free text data in the CBP decision-making process through semantic modeling is limited, leading the CBP's decision-makers to ignore the potential advantages of existing semantic modeling methods when dealing with many free texts. This paper aims to address this gap by introducing knowledge modeling and mining of project introduction text corpus, leveraging semantic technology to support CBP. First, we design an ontology of capability to describe the core concepts relevant to the CBP process. Subsequently, a semantic framework based on the RDF knowledge graph is proposed, enabling humans and machines to comprehend project description texts. To capture the semantic data essential for CBP, a motif structure is employed to model semantic expressions, ensuring their consistency with CBP concepts through compliance checks. Finally, the effectiveness of the proposed semantic framework is evaluated by querying the project’s knowledge graph after semantic normalization, providing an assessment of its potential in CBP applications.

Received: 4 February 2024; Revised: 12 April 2024; Accepted: 12 May 2024.

Keywords: project management, semantics, capability-based planning, knowledge graph, ontology.

1.0 Introduction

Development planning of the national defense field focuses on improving military force and technological development within limited resources. The main goal is to deploy and utilize military means under various conditions to effectively respond to future competitive threats. A well-known approach to achieving this goal is focusing on military capabilities, often called Capability-based Planning (CBP). CBP is a planning framework under uncertainty that provides capabilities suitable for various modern-day challenges and circumstances while working within an economic framework that necessitates choice (Davis 2002). CBP was developed as an alternative to threat-based planning. It brings transparency and consistency to cross-disciplinary technology planning, breaks down barriers in resource allocation within the enterprise, and helps determine the level of resources required to maintain and improve essential capabilities for future competition. Over the past decade, CBP has become the standard for national defense planning across the entire NATO alliance (De Spiegeleire 2011), and the national defense community has widely adopted it.

Although CBP is a mature theory and method, it still faces various challenges in practical application. One significant aspect is that it often involves identifying the highest priority option from many candidate projects, also called Capability Portfolio Management (CPM). The relationship between these candidate projects and capabilities is often implied within free text rather than a structured database. These text data contain valuable knowledge, but understanding this requires highly specialized expertise and is quite challenging for knowledge management. Therefore, the decision-making of the CBP often encounters uncertainty issues: 1) document-based planning leads to ambiguity and difficulties in sharing capability-aspect information within the enterprise; 2) the planning phase and the implementation phase for a capability are taken charge by separate sectors, making it difficult to assess and control the expected outcome; 3) there is a strong dependency between capability increments, and the development plan may have systematic and structural flaws; 4) if defects in the capability development plans are not discovered during the early stages of capability implementation, the cost of remediation is too high to afford.

In the past ten years, the Open Data initiative has become the standard of data sharing on the Internet, which provides the best practice for publishing and sharing linkable structured data (Bizer et al. 2009). Linked data is designed as a technical method to describe knowledge and model the relationship between everything in the world with a graph model. The rapid development of Semantic Web technology provides widely accepted standardized protocols for knowledge modeling and knowledge management in various fields, such as agriculture (Drury et al. 2019), healthcare (Narayanasamy et al. 2022), industrial engineering (Kebede et al. 2022), financial management (Tang et al. 2018), and education (Jensen 2019). The application of knowledge organization systems (KOSs) based on ontology and formal semantics has made it possible to share knowledge on a large scale, thereby facilitating the capturing domain knowledge, assigning semantic meaning to information and representing data for machine consumption (Hjørland 2007, Padmavathi and Krishnamurthy 2017, Ghosh et al. 2020, Smiraglia 2015, Bagchi 2021).

So far, the application of semantic web technology in the semantic knowledge management field of national defense planning is limited. Traditional knowledge modeling methods in this field mainly use Enterprise Architecture (EA) as the framework (Lee and Park 2009, Torkjazi et al. 2022, Martin 2022). People are increasingly interested in using semantic web technology to enhance EA knowledge modeling ability. Hinkelmann et al. (2016) have combined EA modeling with enterprise ontology and proposed using a semantic metamodeling method to address strategic alignment issues. Roach (2011) argues that the existing architecture modeling language is insufficient for capturing the behavior and governance of information systems and suggests using the Resource Description Framework (RDF) as a knowledge modeling tool. In addition, some studies have noticed the technical feasibility of using semantic technology to enhance CBP management (Hoyland 2012, Hoyland et al. 2014, Dibowski et al. 2020). These studies have abandoned EA tools that are difficult to use in practice and used semantic technology as an alternative to traditional EA tools. However, no well-defined ontology resources are available for dealing with tacit knowledge in the free texts, and the CBP management field has not fully utilized the benefit of the linked data.

To solve this gap, we propose an ontology-driven semantic modeling framework for mitigating uncertainty in decision-making in CBP management. Our framework focuses on modeling knowledge in project texts with a compliant semantics alignment in line with requirements like CPM. Ontologies are employed to provide formal knowledge representation, and a motif-based knowledge subgraph structure is used to capture the complex semantics in project texts. By introducing ontologies, unstructured texts can be transformed into machine-understandable knowledge, facilitating semantic compliance processing and offering semantic support for CPM.

The rest of the paper is organized as follows. Section 2.0 provides a background context and motivation for our semantic modeling. Section 3.0 presents the ontology-based framework for capability aspect semantics. Section 4.0 introduces the semantic compliance process for CBP. Section 5.0 conducts experiments for comparison. Finally, Section 6.0 presents the conclusion of the paper.

2.0 A systematic analysis of the CBP decision-making process

In this section, we discuss two questions about our motivation to provide knowledge support to the decision-making process: 1) what problems can semantic knowledge management solve in CBP? 2) How does semantic knowledge management support decision-making based on a semantic method?

CBP is generally a method for planning under uncertainty (Kossakoski 2005). Capability refers to "a business-focused outcome that is delivered by the completion of one or more work packages." According to TOGAF's definition (Papazoglou 2014; Aldea et al. 2015), CBP focuses on planning, engineering, and delivering strategic business capabilities to an enterprise. Many studies have been conducted to effectively utilize existing knowledge to enhance the decision-making for CBP, often involving various concepts related to project management and modeling their relationships, providing semantic compliance in cross-domain design (Martin 2022; Lo et al. 2020). These studies primarily rely on EA tools. However, these methods lack focus on specific data perspectives in these studies. In our research, semantics knowledge is acquired from project texts since conceptual knowledge modeling in CBP's research so far does not adequately incorporate free text. Addressing this gap requires a thorough investigation of concepts and their semantic relationships within specific data contexts.

We analyze this problem from the perspective of CBP's decision-making process. This decision-making process involves multiple steps, from top-level strategic planning to capability planning and specific project engineering. Understanding this process helps us analyze the semantic knowledge the project text data can provide. In the initial stage of strategic planning, decision-makers need to make a strategic selection, which involves an in-depth analysis of the internal and external environment, including evaluating strategic needs, competitive situations, development trends of emerging technologies, and current strengths and weaknesses. This step defines the objectives and goals of long-term development and provides guidance for subsequent decision-making. After determining the strategic objectives and goals, decision-makers need to plan the priority for capability development. This process mainly evaluates whether the existing resources and capabilities meet the demand of strategic objectives and goals. If the existing capability is insufficient, there is a capability gap and a need for development to ensure that the enterprise has available means in the future competition through investment in technology research and development, equipment acquisition, force reform, personnel education, training, and other aspects. Then, specific projects are needed to address the capability gap, which involves decision-making in project selection, investment scale, development roadmap, and risk control. Decision-makers need to ensure that the selected projects are consistent with the overall strategy and that all the capability requirements are satisfied by sufficient resources. At the same time, it is also necessary to consider the project schedule, milestones, and evaluation criteria to ensure that all kinds of projects are implemented on time and on demand and that the promised capability increment is delivered. A complete capability-based strategic planning decision-making process is shown in Figure 1.

The decision-making process shown in Figure 1 reveals that the decision-making of CBP is a project portfolio selection process based on strategic objectives and capability requirements. The decision-maker clarifies the capability requirements in the capability planning phase, while the project engineers provide candidate solutions according to their demand, and the decision-maker then checks and drafts project portfolios from a large number of candidate projects. There are two aspects of planning and evaluation decision-making problems in this process, which may bring cognitive uncertainty.

First, it is the decision-making problem about the capability itself, including why to develop such a capability (evaluation decision) and what kind of capability requirements need to be defined (planning decision). These contents involve decision makers' in-depth understanding and analysis of the internal and external environment, competitive situation, and future development trends. Due to the inherent uncertainty of the external competitive situation, it may be difficult for decision-makers to identify the capabilities that are needed for future battlefields. This uncertainty can also be regarded as the uncertainty of capability requirements, which may lead to difficulties in the capability planning phase and making clear decisions.

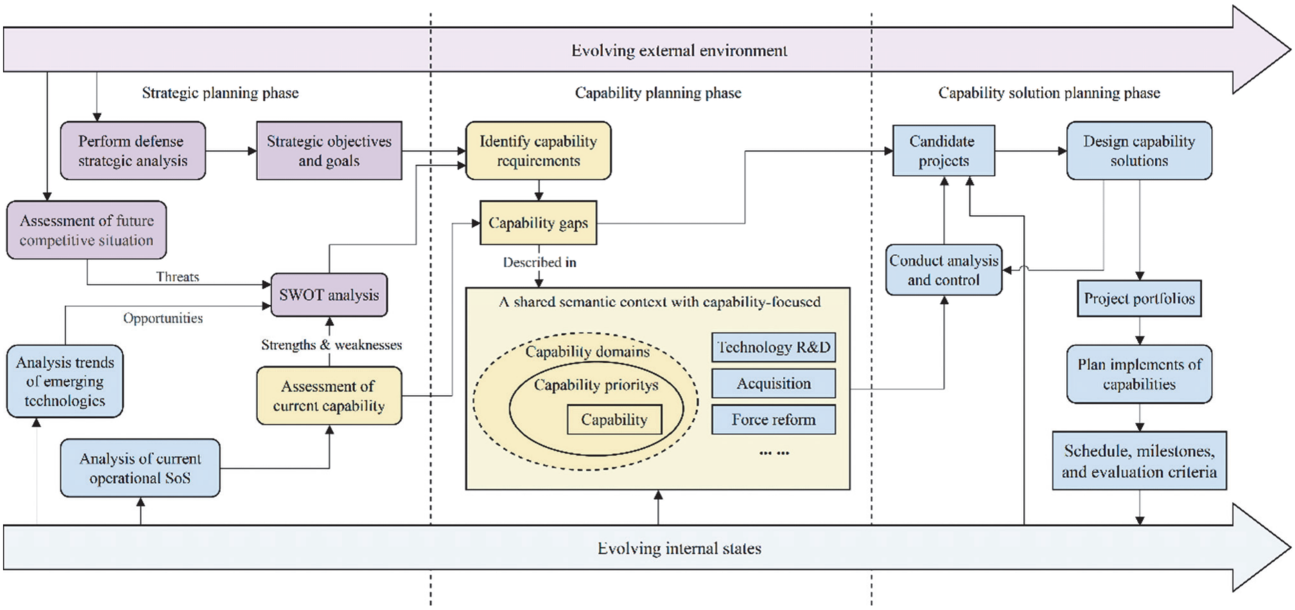

Figure 1: The strategic planning decision-making process of capability-focused.

Secondly, it is about the decision-making problem of the realization of capability, that is, the selection of project portfolio, including how to judge the feasibility of candidate projects meeting the capability requirements (evaluation decision) and which projects to choose as candidates for this capability or capability increment (planning decision). These contents relate to the decision-maker's knowledge of the relationship between a large number of candidate projects and capability or capability increment. Due to the limitations of individual cognition and experience, it is difficult for decision-makers to quickly know the potential capability increment in candidate projects and the potential defects in the selected project portfolio. This uncertainty can be regarded as the uncertainty of capability realization, which may lead decision-makers to miss some vital information when planning.

Based on the analysis of the capability-based planning process, it becomes evident that the decision-making in CBP is derived from two primary sources. The first is the decision-maker's knowledge, while the second relates to the decision-making methodology, which encompasses planning decisions and evaluation decisions. In this context, a structured decision-making process can be perceived as the decision-maker selecting an appropriate model incorporating their knowledge as input. Consequently, the model's output represents the outcome of the decision-making process. Given this perspective, the importance of knowledge in CBP decision-making must be considered, as decision-makers must acquire a comprehensive and accurate understanding of knowledge to mitigate uncertainty's impact on CBP decision-making.

Figure 2 gives a picture of where we expect to integrate linked data into the CBP decision-making process. It shows the possibility of linking the project description's free text with CBP's decision-making activities with semantic knowledge as a bridge, which needs concept modeling to formalize the whole process. Therefore, this study aims to develop an ontology-based framework according to the vision shown in Figure 2, serving the need to mitigate the uncertainty in cross-domain decision-making with a shared semantics context. The challenge involves mining capability semantics from free texts and dealing with the semantic equivalence of different expressions. Traditionally, these semantics need to be explained by experts' knowledge. Our semantic framework will focus on converting free texts of project documents into linked data and provide semantic equivalence transformation. This work will solve the gap between unstructured free texts and formal semantics. Formally modeling this kind of knowledge and making the knowledge data linkable would enhance the automation of CBP decision-making activities, which we believe is a feasible path for semantic knowledge management supporting decision-making.

3.0 An ontology-based semantic framework

The project text encompasses many knowledge entities, making it imperative to design a well-defined and appropriately classified ontology for handling these free texts. Since what we are concerned with belongs to a distinct vertical domain where the classification standards for the open domain are not applicable, it becomes essential to develop a specialized

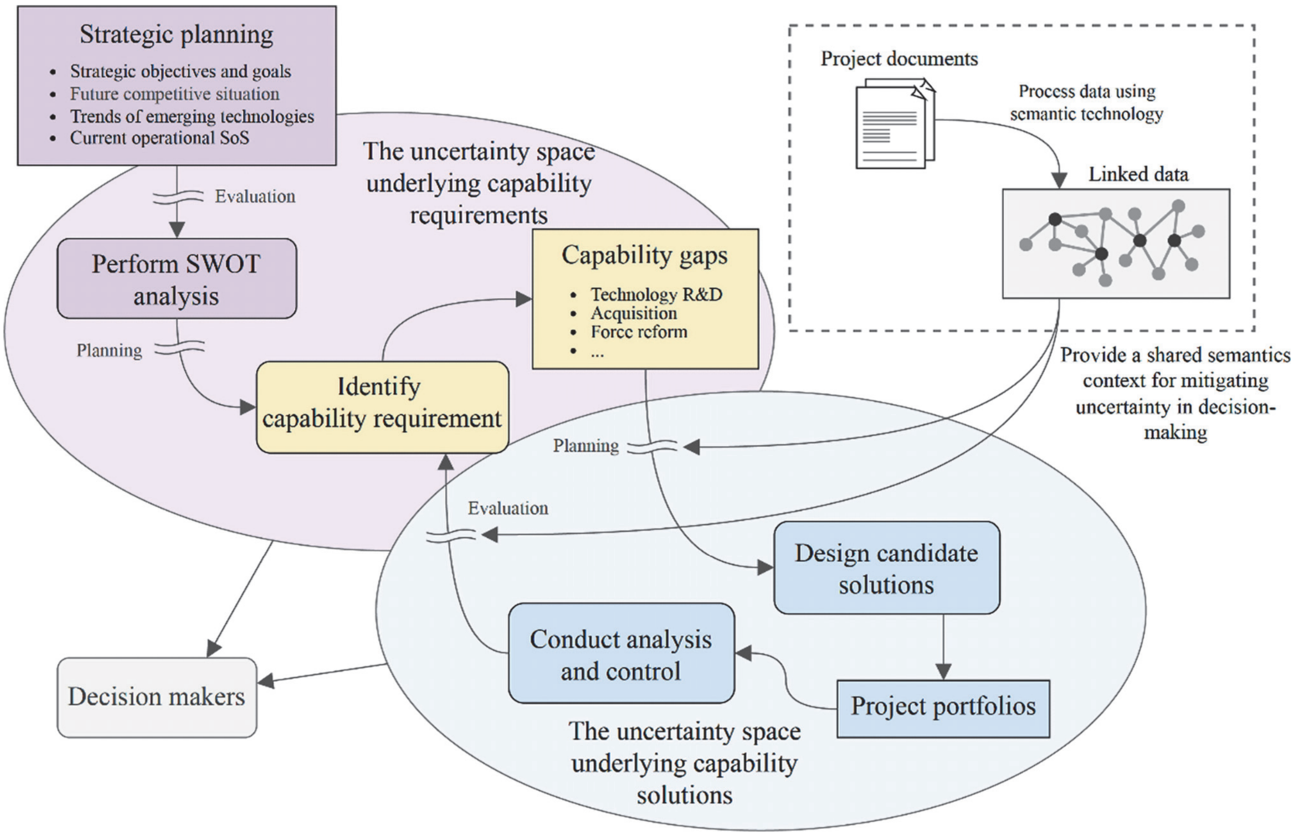

Figure 2: A showcase of incorporating linked data into CBP activities.

ontology that aligns with the characteristics of the studied problem domain. In the subsequent part, we will outline our conceptual knowledge modeling and ontology construction process for the CBP domain, as shown in Figure 3.

3.1 Conceptual knowledge modeling for capability

Ontology engineering is an iterative process, particularly when establishing the initial ontology framework. The knowledge model must account for dynamic adaptability requirements across the cycle of CBP activities. Traditionally, the CBP method uses EA for conceptual modeling and description. Therefore, we propose building an initial ontology based on the existing EA concept models.

EA constitutes a comprehensive depiction of all critical elements and relationships that comprise an organization. An Enterprise Architecture Framework (EAF) acts as a blueprint for describing EA methods (Urbaczewski and Mrdalj 2006). Prominent EAFs like the FEAF (House 2007), TOGAF (Gerber et al. 2010), and DoDAF (Brown 2000) provide abstract descriptions of real-world entities based on metamodels. These metamodels define the abstract concepts and relationships using a structure similar to ontology. We construct an integrated conceptual knowledge structure by drawing inspiration from various metamodels employed in EAFs. The abstract concepts within the ontology are derived from diverse EAF metamodels. They are designed to encompass three layers: the upper-level concept groups, the mid-level ontologies, and the core ontologies.

The upper-level concept groups encompass Capability, Operation, and Mission. These concept groups reflect the domain knowledge that CBP may encompass. Among these aspects, the Capability aspect is the primary focus of our conceptual knowledge modeling. So far, our conceptual knowledge modeling does not include ontologies of operations and tasks. Instead, we have reserved corresponding interfaces for the future expansion of conceptual knowledge modeling in these fields.

The mid-level ontologies consist of three major concept groups in the capability aspect: Capability Solution (CS), Capability Domain (CD), and Capability Motivation (CM). These concept groups bear similarities to the metamodel in EA. For example, the Capability Solution resembles the *Resource* type defined in DoDAF. Major concept groups can significantly reduce the complexity of ontology structures and facilitate consensus-building among CBP stakeholders, thus providing substantial benefits during the iterative and revision process of ontology development.

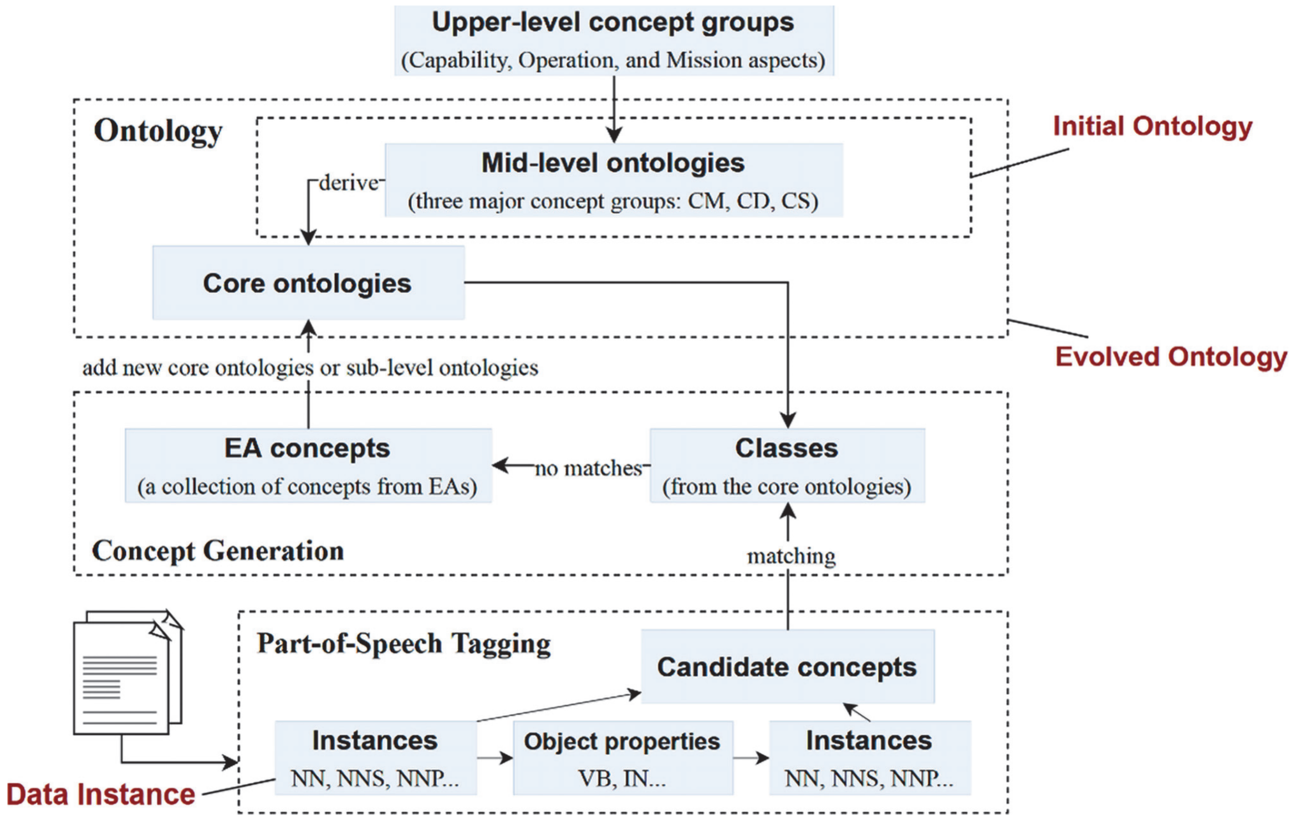

Figure 3: Ontology construction process for CBP.

The design of the core ontology draws heavily on the concept definitions within the metamodel of several major EAs, as well as observations of actual data. As part of the data-driven approach, we extracted project text data as the initial dataset for core concept design. Experts then determined which existing metamodel concepts these text segments should be assigned. We selected high-frequency concepts as our core concepts by analyzing the frequency of assignment for each concept in the metamodel. Table 1 presents the core concepts proposed, with each concept capable of being mapped to concepts within the TOGAF and DoDAF within a specific context.

The use of multi-layer conceptual modeling can provide additional benefits for entity-ontology matching. Given that different experts may have different choices when matching text segments to the core ontology, it is essential to ensure that these segments are unique and unambiguous, particularly in matching top-level and middle-level concept groups. Through multi-layer conceptual modeling, we can introduce flexibility into ontology matching, allowing experts to agree on top-level and middle-level concepts.

3.2 Classes, hierarchies, and properties

The initial ontology design integrates similar concepts, attributes, and examples. The observation and summary of these elements form an initial ontology relevant to the domain. To simplify conceptual modeling, we aim to integrate similar concepts and attributes in the initial ontology. For instance, when modeling Users, both "Organization" and "Role" can be seen as concrete forms of the abstract concept of "User". Our conceptual knowledge modeling is based on the concepts and abstract syntax of Resource Description Framework Schema (RDFS) and Web Ontology Language (OWL) (McGuinness 2004), which allow for the specification of instances as subclasses of a class or the classification of an instance by an attribute. Therefore, by designating *cbp:Organization* or *cbp:Role* as a subclass of *cbp:User*, various types of entities will be unified into the concept of "User".

The initial ontology design also considers their semantic relationships, such as *cbp:develop* and *cbp:support*, which stems from expert consensus in the CBP domain. For example, the *cbp:develop* relationship reflects the semantic connection between a project and its primary products, which usually consist of a system or technology or may also include other research outcomes like design, software, reports, etc.

Concepts in our ontology		Corresponding concepts in EAFs	
Core	Sub-level	DoDAF	TOGAF
Technology	-	-	Technology Component, Technology Service, etc.
System	Sub-system	System	Information System
	Component	System	Application Component
Capability	Enterprise Capability	Capability	Capability
	Mission Capability	Capability	Business Capability
	System Capability	Capability	-
Performance	-	Measure	Measure, Service Quality, etc.
Function	Service	Service	Function, Service, Business Service
Mission Need	-	Vision	Driver, Requirement, Course of Action, etc.
Environment	-	Condition	Constraints
Project	Program	Project	-

Table 1: Ontology design and its correspondence with EAFs.

We aim to minimize distinctions and merge similar semantic relations whenever possible when designing semantic relationships. Additional semantic relationships will be expanded according to the actual data mining situation.

In summary, our goal is to maintain simplicity in ontology design. Whenever a new concept needs to be modeled, we first explore whether it can be expressed using defined concepts and attributes. If necessary, we extend the design by modifying the definition of attributes within the original ontology rather than generating new concepts.

3.3 Core ontologies

Figure 4 provides a visual representation of our ontology conceptual model, highlighting the key entities and relationships within the core ontology. Here, we summarize the fundamental ideas guiding the design of the conceptual model for the core ontologies.

1) To guide the grouping of concepts within the core ontology, we utilize three mid-level ontologies: *cbp:CS*, *cbp:CD*, and *cbp:CM*. Each group consists of ontology concepts combined with semantic structures that capture the specific details of the capabilities associated with CBP.
2) *cbp:CD* (Capability Domain): We distinguish between two forms of expression, one as an abstract description of the capability and the other as a structured capability with more comprehensive details. The abstract description of capabilities can be defined by structured capabilities and connected using *rdf:isDefinedBy*. Additionally, we define three subclasses of capability: *cbp:EnterpriseCapability*, *cbp:MissionCapability*, and *cbp:SystemCapability*. These subclasses describe different levels of capabilities.
3) *cbp:Project*: This entity is the central element of our ontology. A project's text consists of the project name and referential information, such as a text segment like "This Project" that references the project. We use the project name as the primary entity and other referential information is connected to the associated project using *rdfs:sameAs*.
4) *cbp:CS* (Capability Solution): The capability solution comprises two sub-classes, namely *cbp:Technology* and *cbp:System*, which form the main components of project development. The project ontology, *cbp:Project*, is linked to *cbp:CS* through the relationship *cbp:develop*.
5) *cbp:CM* (Capability Motivation): Capability Motivation encompasses the needs and motivations driving project development, including several sub-classes. To guide the evolution of the ontology, we have defined initial semantic relations for these subclasses. These relationships, which include *cbp:satisfy*, *cbp:concern*, *cbp:support*, and *cbp:enhance*, are classified based on their associated tail entity classes.
6) *cbp:Environment*: Environment refers to restrictive conditions that impact Capability Motivation and Capability Domain.

4.0 A semantic compliance process using SPARQL

Here, we present the semantic compliance process based on the initial ontology discussed in Section 3.3.

4.1 Overview

The semantic compliance process, which includes three stages, is shown in Figure 5. The process starts with a data-driven entity relationship recognition process and progressively expands our ontology framework based on the actual

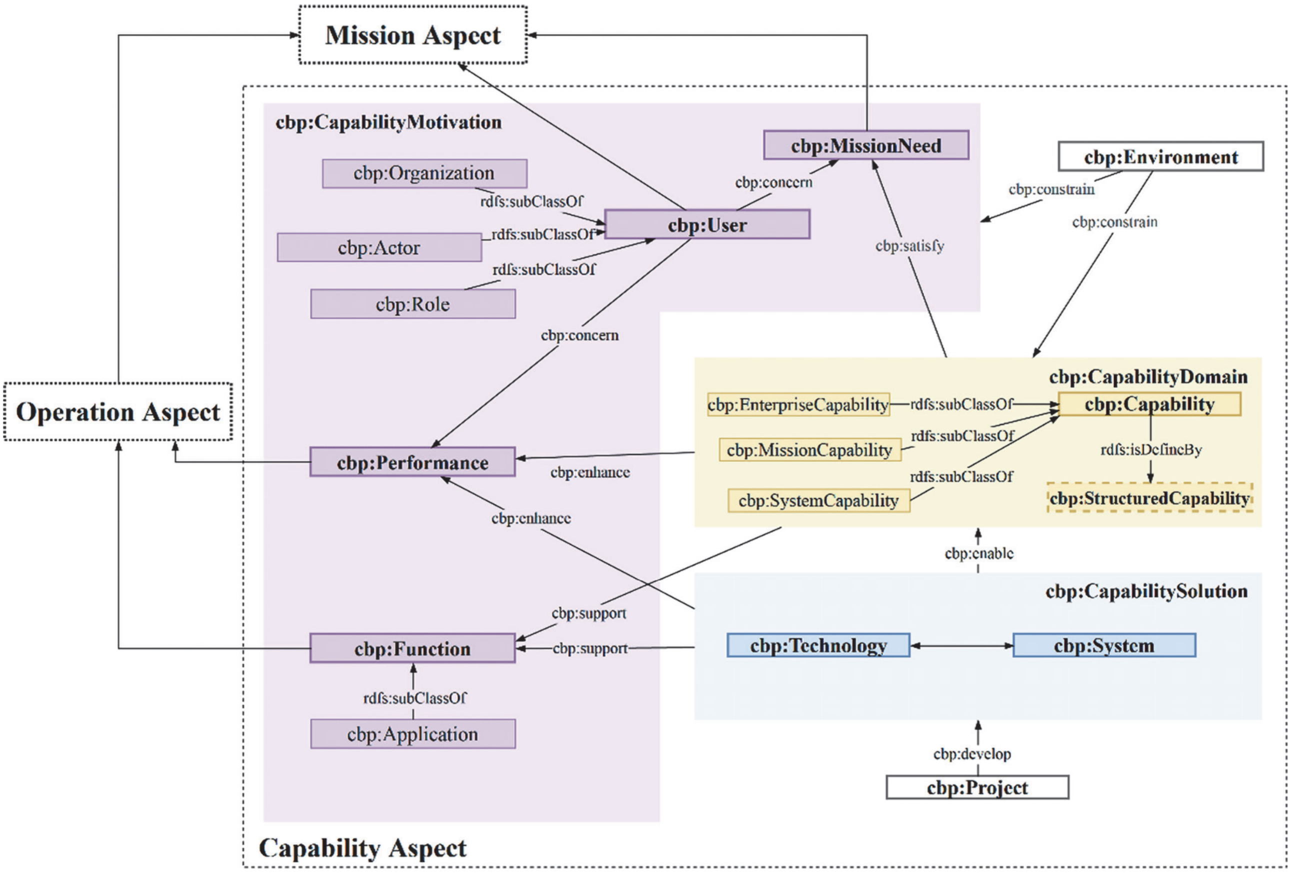

Figure 4: A visual CBP ontology based on OWL.

text data. To achieve this, we designed a text analysis method and algorithm that relies on the dependency of the text. By identifying text segments and matching them with ontology concepts using keywords and expert knowledge, the semantics within the text are formalized. A motif-based structure comprising two triples presents the N-arity semantic expression structure, automatically mining the most common forms of entity relationships in the text data. The semantic relations are mined by identifying frequently occurring semantic expression structures. Finally, using SPARQL, the mined semantic data are transformed into a compliant format according to the CBP decision-making context.

4.2 Mining semantics from texts

Given that the project text contains numerous characteristic words and proper nouns, it is crucial to ensure that the extracted information has clear semantic boundaries. Therefore, this stage employs semantic segments mined from sentences as the minimum information units. These segments are then used to construct a graph-based database.

Sentence segmentation is a technique that divides a document into smaller parts, referred to as segments. Based on the spaCy library (https://spacy.io/), a text processing program was developed to identify dependency and extract segments. SpaCy provides a method to merge noun phrases (NPs) before text processing by incorporating the “merge_noun_chunks” parameter in its pipeline, which greatly simplifies the extraction of NPs. However, this approach has limitations when dealing with complex NP structures. An example is given in Figure 6. In this case, we use a segment merging algorithm based on line segment sorting to get non-repetitive and non-overlapping segments, as shown in Algorithm 1.

Algorithm 1 Segment merging

Require: A list of NPs $S = \{s_i\}$, $i = 1, \ldots, n$, each NP $s_i = (start_i, end_i)$. Here, $start_i$ and end_i represent the starting point and end point of s_i.

$S_{sorted} \leftarrow$ sort S according to $start_i$, where $S_{sorted} = \{\hat{s}_j\}, j = 1, \ldots, n$

$S_{merged} = [\hat{s}_1]$

for each element $\hat{s}_j$ in S_{sorted} **do**

  $\hat{S}_t \leftarrow$ the last element in S_{merged}

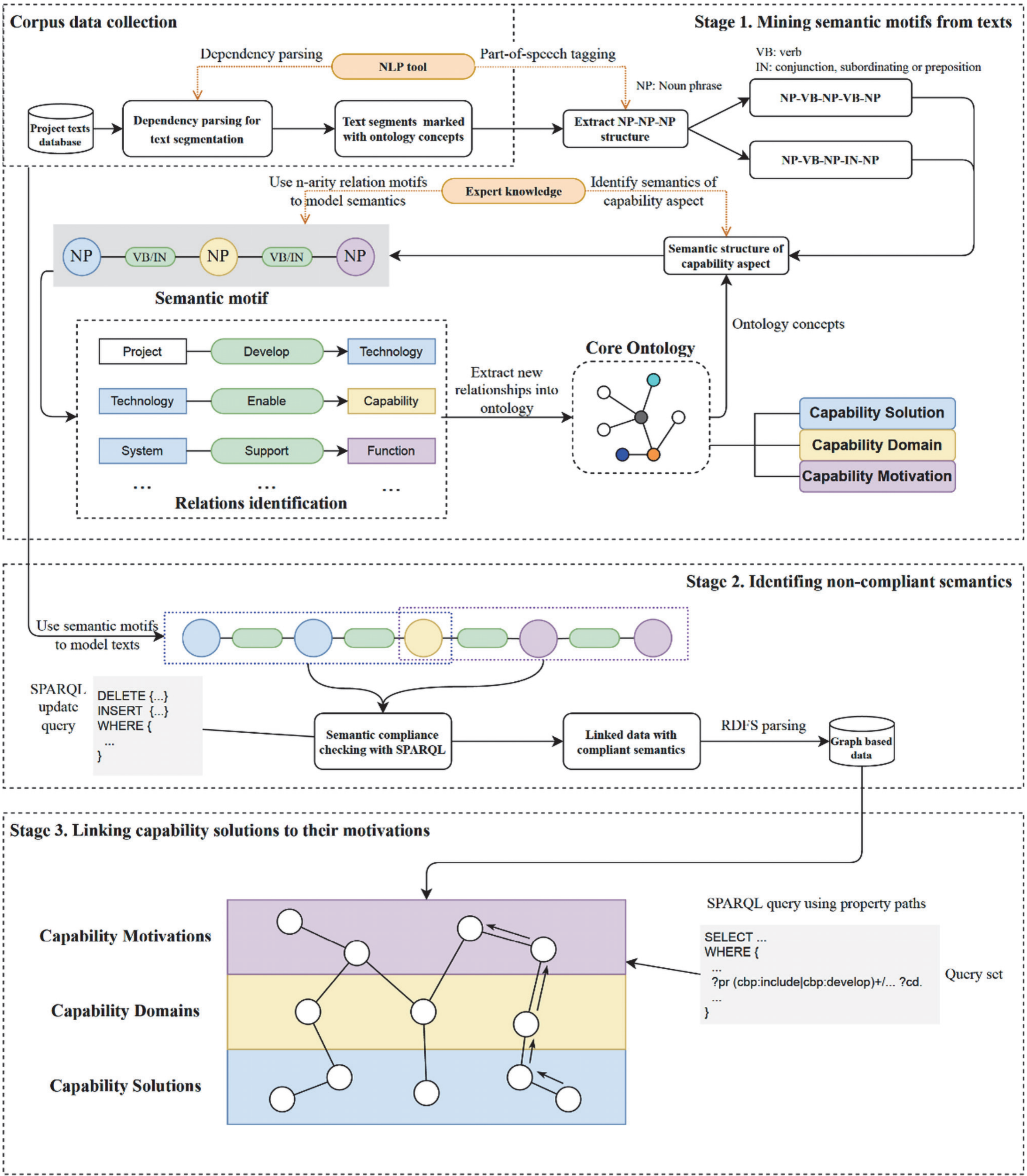

Figure 5: A semantic framework used to extract and mine the semantics of capability aspect in texts.

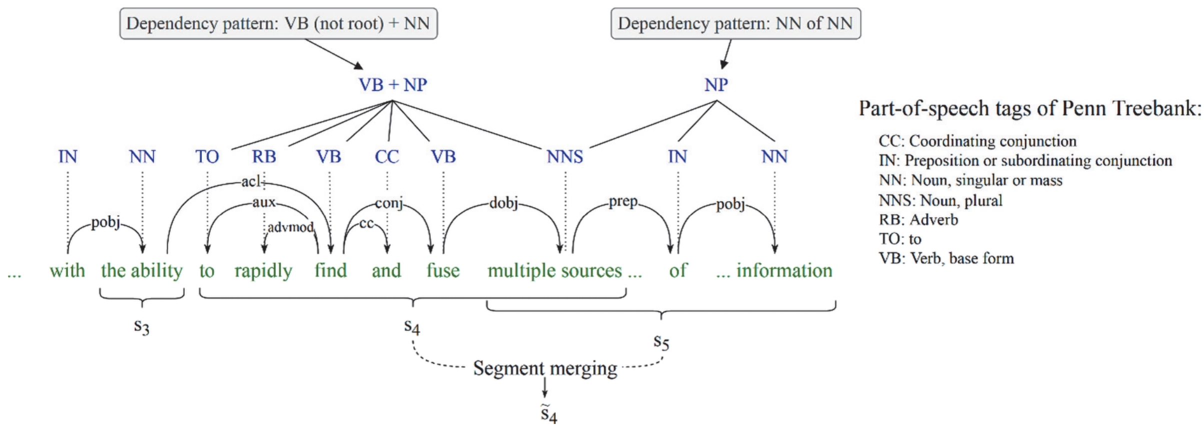

Figure 6: Identify NPs through part-of-speech tagging.

```
        if start_j ≤ end_t then
            if end_j > end_t then
                end_t ← end_j
            else
                pass
            end if
        else
            add ŝ_j to the end of S_merged
        end if
    end for
    return S_merged
```

Based on experts' knowledge, each sentence's NPs are assigned a core ontology, allowing us to extract relevant texts and explore the relationship categories between these core ontologies. The structure of partially extracted sentences is presented in Table 2, where the text segments of NPs are replaced with their respective core ontology concepts.

Typically, a triple structure like <*subject*, *predicate*, *object*> can be used to depict the fundamental relationship between entities. For example, Miao et al. (2020) employ a TRT structure consisting of Technology (noun phrase), Relationship (preposition phrase), and Technology (noun phrase) as the underlying framework for semantic analysis, enabling the extraction of relationships between technical terms in patent texts. However, in many cases, the expressive capability of triples needs to be improved when attempting to capture complex relations. Consider a semantic structure "*Pr evaluates S for S*" and its two triples in ontological form:

$$< Pr, evaluates, S >, < S, for, S > \quad (1)$$

Obviously, the triple < *S*, *for*, *S* > can not give meaningful semantics and may lead to confusion in understanding. Consequently, it is necessary to employ complex semantic expression structures to model capability-focused semantics.

To address the issue of insufficient representation capability exhibited by a single triple, we use network motifs as the basis for solving this problem. Network motifs are higher-order structures that are small network sub-graphs (Benson et al. 2016). Formally, given a connected sub-graph $G = (V, E)$ representing a sentence containing n semantic segments and an ontology set $O = (Class, Property)$. Based on the order of the segments in the sentence, each adjacent node pair v_i, v_j has a connection edge $e_{ij} \in E$. For a sentence with at least three semantic segments, we can define a motif m with three nodes v_i, v_j, v_k on its graph G through their ontology set $O = (Class, Property)$:

$$m = < c_i, p_{ij}, c_j, p_{jk}, c_k > \quad (2)$$

where $c_{i,j,k} \in Class$ is the class of the nodes v_i, v_j, v_k, and $p_{ij,jk} \in Property$ is the property type of e_{ij}, e_{jk} defined in the ontology.

As shown in Figure 7, we mine motifs by constructing a connected subgraph G^* of a sentence based on the dependency relations among semantic segments. This is grounded on the assumption that any lexeme can be connected to the ROOT of the sentence through a finite number of dependency relations. Each semantic segment (including NP nodes v and connector nodes p) is numbered sequentially, starting from 0 and connected to the ROOT according to dependency relations. For some dependency relations, e.g. "conj," "advcl," their connection relationships need to be adjusted to ensure that connector nodes p are only connected to NPs nodes v. Subsequently, by converting connector nodes into edges, the directed graph G^* is transformed into a graph G containing only NPs nodes for motif mining. Let $\mathbf{A} = (a_{ij})$

Example sentences and their semantic structures
Example 1: “This project develops combined/advanced cycle air-breathing high-speed and hypersonic propulsion technologies to provide revolutionary propulsion options for warfighters.” Semantic structure: *Pr develops T to provide F for U*
Example 2: “This project evaluates lubricants, mechanical systems, and combustion concepts for advanced turbine engines, pulse detonation engines, and combined cycle engines.” Semantic structure: *Pr evaluates C for S*
Example 3: “This project also develops technologies to increase turbine engine operational reliability, durability, mission flexibility, maintainability, and performance while reducing weight, fuel consumption, and cost of ownership.” Semantic structure: *Pr develops T to increase Pf*
Example 4: “This project develops component technology for an adaptive cycle engine architecture that provides both optimized performance and fuel efficiency for widely varying mission needs.” Semantic structure: *Pr develops T for F that provides Pf for M*
Example 5: “This project evaluates hydrocarbon-based fuels for legacy and advanced turbine engines, scramjets, pulse detonation and combined cycle engines.” Semantic structure: *Pr evaluates S for S*
Symbolic meanings: *C*-Capability, *E*-Environment, *F*-Function, *M*-MissionNeed, *Pf*-Performance, *Pr*-Project, *S*-System, *T*-Technology, *U*-User.

Table 2: Examples of the semantic structure of sentences

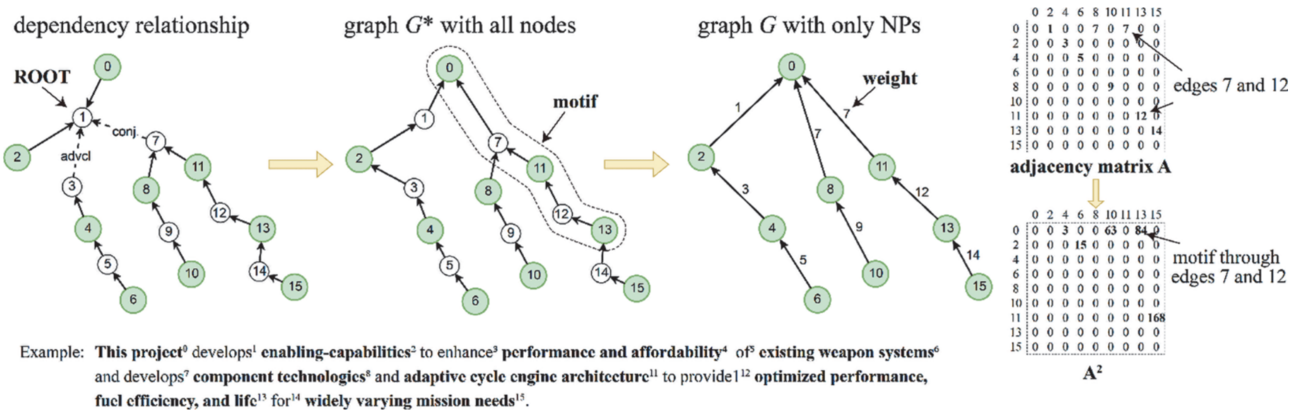

Figure 7: Mining motifs based on weighted adjacency matrix.

be the weighted adjacency matrix of G, where a non-zero element $a_{ij} = p$ indicates that v_i and v_j are connected through the p-th connector node. Since there are no loops in graph G, there is only one way to connect any three nodes. Therefore, a motif exists among nodes v_i, v_j, and v_k if and only if $a_{ij}a_{jk} > 0$. In this case, the element in the matrix $\mathbf{A}^2 = (\alpha_{ik})$ represents the product of the connector p and q connecting nodes v_i, v_j, and v_k:

$$\alpha_{ik} = \underbrace{a_{i1}a_{1k} + \cdots}_{0} + a_{ij}a_{jk} + \underbrace{\ldots + a_{in}a_{nk}}_{0} = \begin{cases} pq, \text{motif exists between } i, j, \text{and } k \\ 0, \text{motif not exist between } i, j, \text{and } k \end{cases} \qquad (3)$$

After mining motifs by Algorithm 2, we summarize the most prevalent types of motif m mined from free text and use them to enhance the semantic extraction. This process is also used to complete the missing property types of the ontology design in Section 3.3. These property types based on free text mining are shown in Table 3. We provide our code here: https://gitee.com/rs023/preTech2.

Algorithm 2 Motifs mining

```
Require: S, the list of all segments (ontology-format) from a sentence, S = {s_i}.
         G, the sub-graph of all NPs in S, A is the adjacency matrix of G.
for each element α_ik in A^2 do
if α_ik > 0 then
    r ← the i-th row of A
    c ← the k-th column of A
    e = r · c
    j ← the index of non-zero element in e (only one)
    p ← the j-th element in r
    q ← the j-th element in c
    motif = {s_i, s_p, s_j, s_q, s_k}
    add motif to M
return M
```

4.3 Identifying non-compliant semantics

Using a motif-based structure, we analyze the text information mined to determine what motif structure forms can express semantics related to the capability aspect. As we continue to mine data, more motif structures about capability aspects are being discovered. Table 4 shows these candidate semantic motifs according to the definition of ontology.

Obviously, these motif structures need further examination to provide correct semantic information equivalent to the original text. For instance, the correct semantic expression in Example 3 should be modeled by motifs:

$$Pr\ develops\ T\ in\ E\ for\ C \rightarrow \begin{cases} < Pr, develop, T, constrainedBy, E > \\ < \mathrm{Pr}, develop, T, enable, C > \end{cases} \quad (4)$$

To address the above semantic equivalence issue, we propose using customized motif templates to transform the free semantics into a unified form. Generally, we want to establish these semantic motifs in a form similar to "*cbp:CS*→

Relation	Domain	Range	Inverse relation	Common words
cbp:linkTo	-	-	*cbp:linkFrom*	-
cbp:develop	*cbp:Project*	*cbp:CS*	*cbp:developedBy*	develop, investigate, demonstrate, focus on, evaluate, etc.
cbp:enable	*cbp:Project/CS*	*cbp:CD*	*cbp:enabledBy*	enable, achieve, deliver, assess, etc.
cbp:support	-	*cbp:CM*	*cbp:supprtedBy*	support, ensure, for, etc.
cbp:enhance	-	*cbp:Performance*	*cbp:enhancedBy*	improve, enhance, etc.
cbp:include	-	-	-	include
cbp:associate	-	-	-	of, for, in, etc.
cbp:constrain	*cbp:Environment*	-	*cbp:constrainedBy*	in
cbp:concern	*cbp:User*	-	*cbp:concernedBy*	focus, critical to

Table 3: Relations associated with capability semantic.

Example sentences and their candidate motifs	
Example 1: "This project develops ... technology for ... architecture that provides ... efficiency for ... needs."	
Semantic structure:	*Pr develops T for F that provides Pf for M*
Candidate motifs:	<*Pr*, *develop*, *T*, *support*, *F*>, <*T*, *support*, *F*, *enhance*, *Pf*>, <*F*, *enhance*, *Pf*, *support*, *M*>
Example 2: "The enabling technologies developed under this project will be used for ... capabilities."	
Semantic structure:	*T developed under Pr will be used for C*
Candidate motifs:	<*T*, *developedBy*, *Pr*, *enable*, *C*>
Example 3: "This project develops ... technologies in ... environment for ... capabilities."	
Semantic structure:	*Pr develops T in E for C*
Candidate motifs:	<*Pr*, *develop*, *T*, *constrainedBy*, *E*>, <*T*, *constrainedBy*, *E*, *linkTo*, *C*>

Table 4: Examples of candidate semantic motifs mining.

cbp:CD→cbp:CM". Identifying equivalent semantic forms in motif structure is the key step to realizing semantic compliance. This requires establishing some rules and transforming the semantic motif to compliant triples based on these rules, as shown in Figure 8.

There are many ways to realize the above functions, such as checking the semantics of linked data and adding new semantic links through reasoning language based on the Description Logic (DL). In practice, we find that the graph update operations provided by the SPARQL language can achieve the same goals more efficiently and directly. The specific method is to define the query rules through the WHERE keyword and then modify the graph through the DELETE and INSERT keywords. An example of the query is shown as follows:

```
DELETE {?environment cbp:enable ?cd}
INSERT {?cs cbp:enable ?cd}
WHERE {
 ?cs cbp:constrainedBy ?environment.
 ?environment cbp:enable ?cd.
 ?cs rdf:type cbp:CS.
 ?environment rdf:type cbp:Environment.
 ?cd rdf:type cbp:CD
}
```

4.4 Linking capabilities to their motivations and implementations

Once the connections of semantic segments are updated through the compliance process, our focus shifts to making

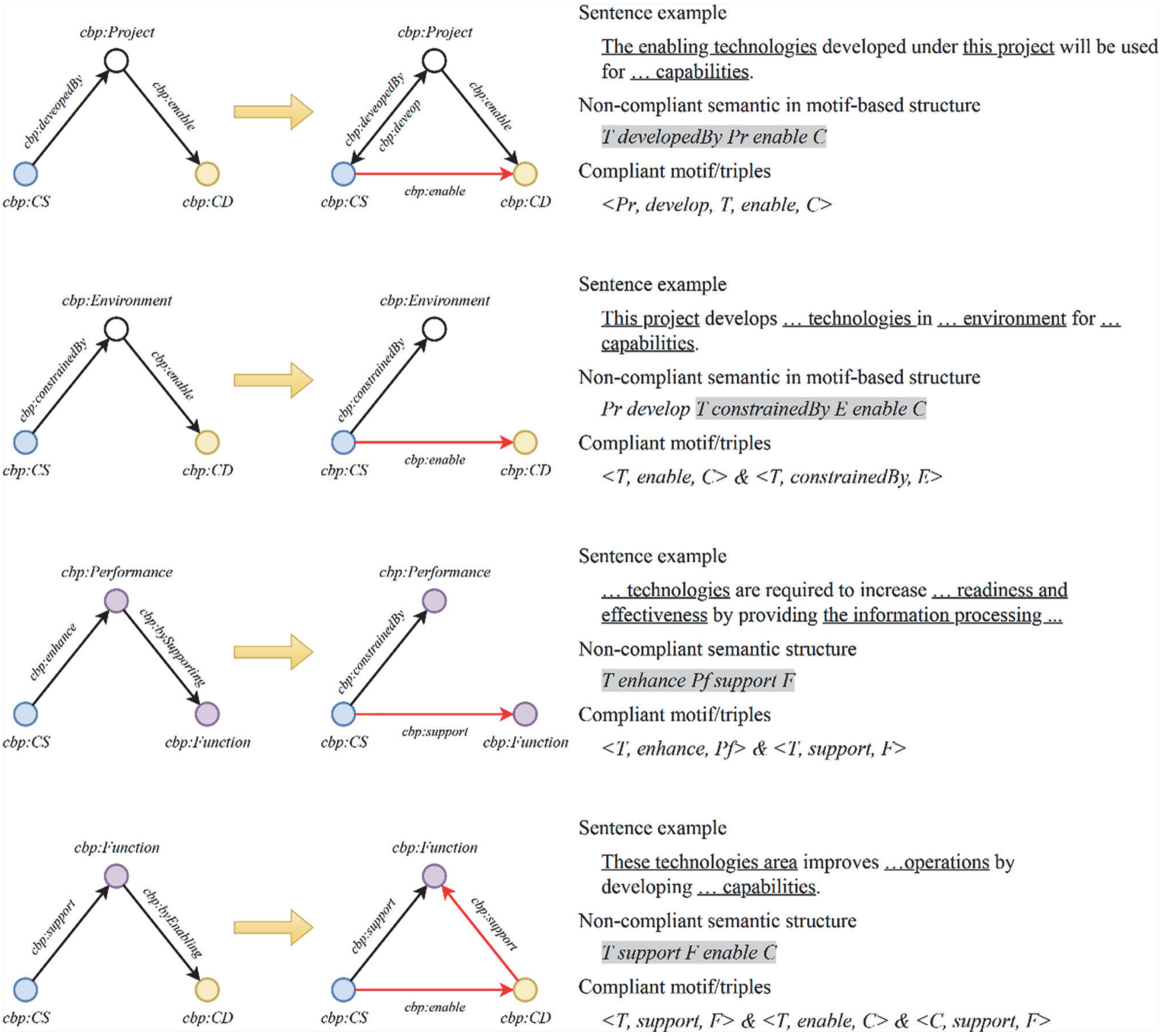

Figure 8: Semantic compliance checking for the capability aspect based on a motif structure

the linked data available for the decision-making application shown in Figure 2. The knowledge database consists of linked data that provides a means of linking capabilities with their corresponding motivations and implementations, giving convenience for decision-makers to query capability-focused semantics. To this end, we utilize the SPARQL language and query based on our compliant motif-based graph data. SPARQL language allows for straightforward querying using property path expression, which aligns with our objective.

A property path expression in SPARQL is similar to a string regular expression but over properties, not characters. Query evaluation determines all matches of a path expression and binds the subject or object as appropriate. The syntax of property paths querying in SPARQL can comprise the elements listed in Table 5.

Based on semantic compliance processing, a simple SPARQL query statement that finds the details of the capability solution in the database (denoted as ?x) and links with their motivations can be expressed as:

```
SELECT ?cs ?x ?cm
WHERE {
 ?cs cbp:include ?x.
 ?cs cbp:support ?cm.
 {
  SELECT ?cs
  WHERE{
   ?pr rdf:type cbp:Project.
   ?pr (cbp:develop|cbp:include)+ ?cs.
  }GROUP BY ?cs
 }
}
```

5.0 Experiments

This section assesses the ontology-based semantic framework's utility and exemplifies its intended usage. To illustrate how the semantic compliance process works, we provide an example text and two graph databases for the experiment. A set of open-source tools are used to provide our ontology-based and graph-based linked data experimental environment, including:

1) RDFLib (https://rdflib.readthedocs.io/): a Python library for working with RDF, providing Parsers and Serializers for generating RDF data in RDF/XML, N3, NTriples, N-Quads, Turtle, and other formats. We developed a program based on RDFLib to generate RDF format triples. It helps us generate RDF data with ontologies automatically in an environment integrated with text processing.
2) Protégé (Knublauch et al. 2004): This ontology editor supports the OWL and RDF specifications, providing a plug-and-play environment. Its visual plug-in shows our RDF data, and a plug-in of Semantic Web Rule Language (SWRL) (Horrocks et al. 2004) helps add the OWL-based semantic relations in triple data before the semantic compliance processing.
3) SPARQLWrapper (https://sparqlwrapper.readthedocs.io/): a Python library that performs queries remotely by wrapping the SPARQL service. It helps by creating SPARQL-based query invocation and, optionally, converting the result into a more manageable format.
4) Fuseki (https://jena.apache.org/): a SPARQL server of Apache Jena, providing SPARQL endpoint for querying and managing triple data. We use it as a server to answer SPARQL queries in SPARQLWrapper over HTTP.

The above tools constitute a simple integrated environment to realize our framework, as shown in Figure 9. The experiment first checks the logical correctness of our semantic compliance process through a simple text example and then compares the query performance of compliance/non-compliance databases by setting two query scenarios. A SPARQL query test set is proposed to test our semantic compliance process, given in Section 7.0. It contains ten SPARQL queries, including different query forms and functions, to demonstrate our

Syntax form	Matches
(elt)	A group path elt, brackets control precedence.
elt1/elt2	A sequence path of elt1, followed by elt2.
elt1|elt2	An alternative path of elt1, or elt2 (all possibilities are tried).
*elt**	A path of zero or more occurrences of elt.
elt+	A path of one or more occurrences of elt.
elt?	A path of zero or one elt.

Table 5: The syntax of property paths querying.

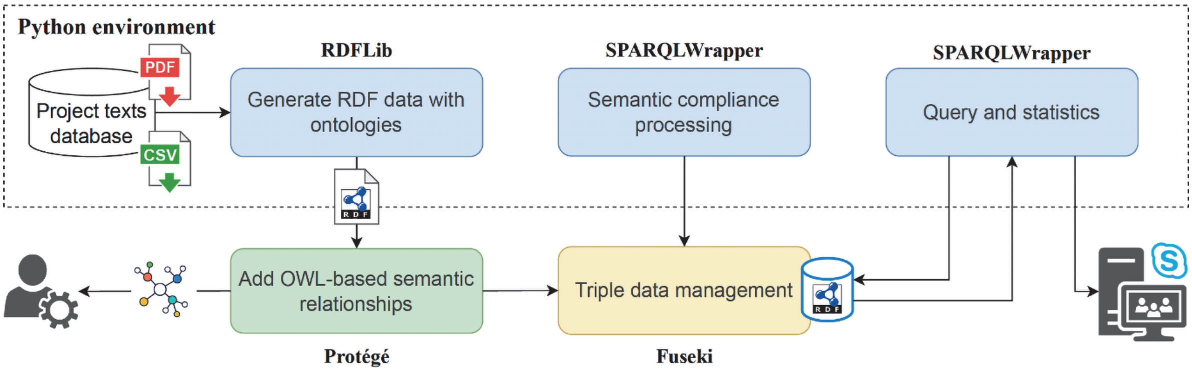

Figure 9: The experimental environment.

Ontology	Text
cbp:Project	"This project"
cbp:Technology	"the required navigation technologies"
cbp:Function	"generating, controlling, receiving, and processing electronic and photonic signals"
cbp:Environment	"severe weather"
cbp:Performance	"the survivability of aerospace vehicles"
cbp:Capability	"intelligence, surveillance, reconnaissance, and precision positioning capabilities"
cbp:System	"exploratory electronic and optoelectronic devices, components, microsystems and subsystems"

Table 6: The semantic segments of the example text.

framework's feasibility of using SPARQL to realize semantic query and reasoning. Queries 1 to 6 realize the semantic compliance checking process according to Section 4.0. Queries 7 to 10 are used to test the query performance for two databases.

5.1 Semantic compliance processing

First, an example text is selected for the experiment: "This project focuses on the required navigation technologies that support generating, controlling, receiving, and processing electronic and photonic signals in severe weather to enhance the survivability of aerospace vehicles. The enabling technologies developed under this project will be used for intelligence, surveillance, reconnaissance, and precision positioning capabilities. The technologies developed include exploratory electronic and optoelectronic devices, components, microsystems, and subsystems."

Based on Algorithm 1, a Python program first segments the text and generates graph data in RDF format. The text is divided into 7 semantic segments and given an ontology, as shown in Table 6. The graph data is visualized in Protégé (Figure 10).

Queries 1 to 6 are used to query the graph data for semantic compliance processing. As shown in Figure 11, the semantic structures of the text segment are reconstructed from Type-1 to Type-2 after semantic compliance processing (Table 7).

Semantic structures	
Type-1	*Pr develop T support F constrainedBy E enhance Pf* *T developBy Pr enable C* *T include S*
Type-2	*Pr develop T support F constrainedBy E* *F enhance Pf* *Pr develop T enable C* *T include S*

Table 7: Semantic structures before and after semantic compliance processing.

5.2 Query performance comparison

This section shows how our semantic compliance processing improves the query with a specific purpose for CBP decision-making by SPARQL's property paths querying. To

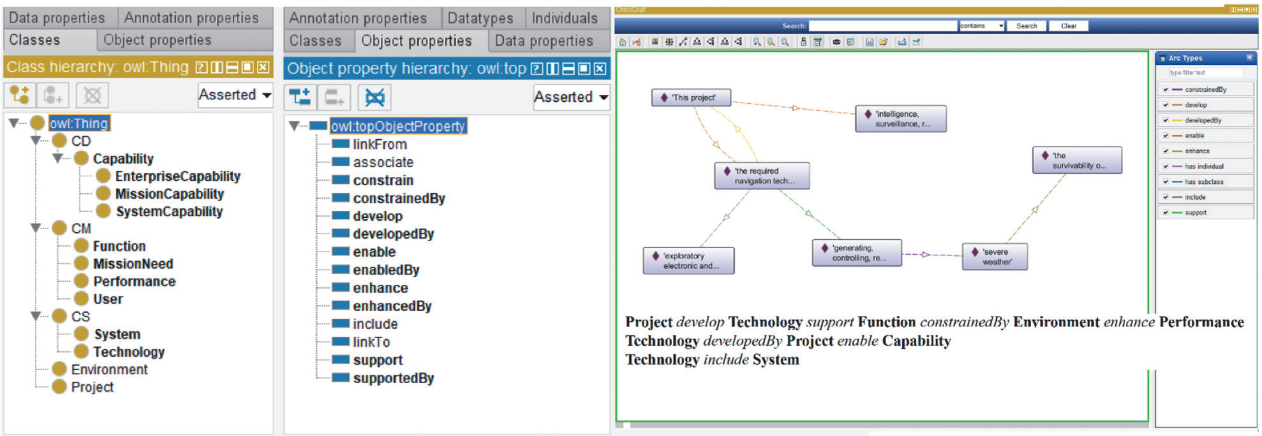

Figure 10: OWL ontology definition and entity relations visual in Protégé.

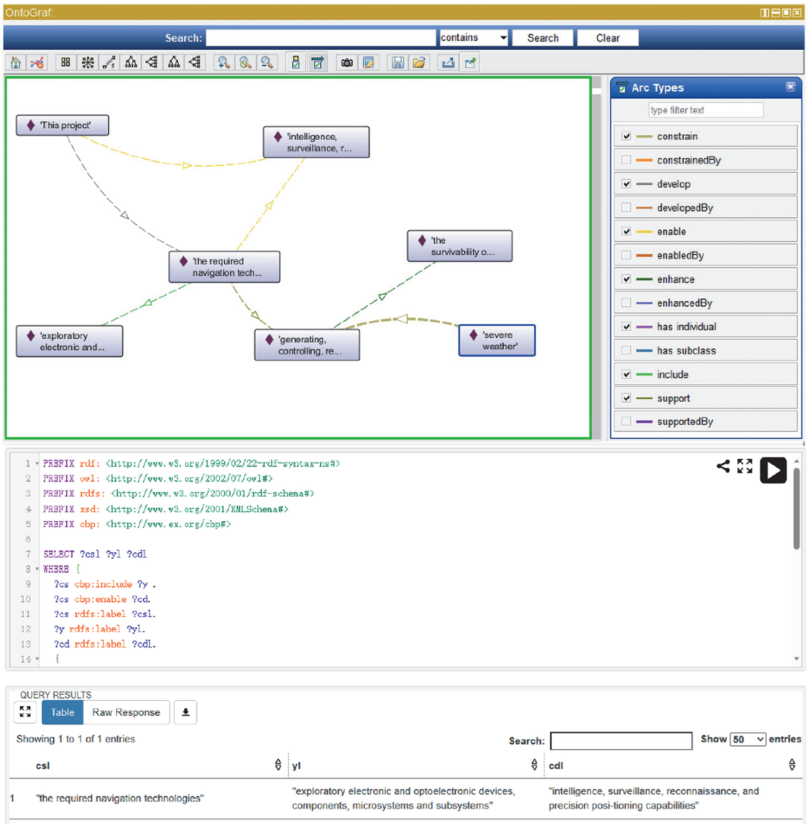

Figure 11: Visualize Type-2 semantic structure in Protégé and query compatible triples in Fuseki.

this end, a graph database *GDB* generated by 2,000/4,000 documents with a similar semantic structure is used as the test set. These documents contain sentences with a similar semantic structure as the above example, and we use them to test the impact on the query performance before and after the semantic compliance process.

Specifically, *GDB* is constructed with both Type-1 and Type-2 semantic structures to simulate linked data in practical applications, as shown in Table 7. The two semantic structures have the same semantics, but Type-2 complies with the semantics specified by our ontology, while Type-1 is non-compliant. To verify the effectiveness of our method, another graph database, GDBc, with semantic compliance processing, is used for comparison, which only has a Type-2 structure, as shown in Table 8.

Database	Semantic structures	Storage sizes	
		Docs	Triples
GDB	Type-1 + Type-2	1,000 + 1,000	60k
		2,000 + 2,000	120k
GDBc	Type-2	2,000	60k
		4,000	120k

Table 8: Database for experiments.

First, two scenarios according to the decision-making of CBP are set in our experiments:

1) Scenario 1: Query descriptions refer to improving some performance in a specific environment, which needs linking between *cbp:Project* and *cbp:Performance*.
2) Scenario 2: Query implementation details of capability solutions related to a specific capability, which needs to query the entities linked to *cbp:CS* through the *cbp:include* relationship.

Two groups of queries with the same purpose in Section 7.0 are used to query *GDB* and *GDBc,* respectively. Queries 7 and 8 are used for scenario 1, and Queries 9 and 10 are used for scenario 2. For *GDB*, since different semantic structures are contained, all possible situations must be considered when querying. Queries 7 and 9 use SPARQL's UNION keyword to achieve this goal. For *GDBc*, we use Query 8 and 10 to achieve the same function.

A Python program that uses the SPARQL wrapper library is built to query the constructed dataset. The performance is evaluated by measuring the program's response time. In this context, the response time is defined as the duration in seconds between when a consumer program initiates a SPARQL query and when it receives the query result from Fuseki's SPARQL endpoint. To eliminate network latencies, the test was run locally. Execute the query set 100 times on the test data to simulate the real usage scenario. The average response time of each query is shown in Figure 12.

6.0 Conclusions

Our research proposes an ontology-based semantic framework to organize and manage text data from project documents efficiently and aims to mitigate the cognitive uncertainty in the decision-making process from the perspective of semantic compliance. Our data-driven method uses the ontology-based framework to model a wide range of complex knowledge related to capability semantics. This enables the construction of a knowledge base for capability-based project portfolio management. Ontology also enables the use of a motif-based subgraph structure to manage the semantic equivalence problem of free text. The semantic compliance issues in free text expression are solved using the SPARQL query, reducing the uncertainty of knowledge in decision-making.

The preliminary experiment demonstrates that leveraging data-driven approaches and semantic web technology facilitates the integration of significant semantic knowledge from free texts of candidate projects into a knowledge base. This tacit knowledge is further standardized by semantic compliance processing, and the compliant semantic linked data are consistent with our ontology, which provides convenience for decision-makers to understand the data structure in the knowledge base. To prove the other benefits of semantic compliance processing, we use a compliant/non-compliant graph database to demonstrate their differences in query scenarios. The graph database with semantic compliance has apparent advantages in query generating and response speed, which helps our expected application for CBP.

Our research has limitations. One limitation lies in the speed of semantic compliance, and we have yet to conduct a comparative analysis of semantic compliance implementation methods based on extensive data. Considering the semantic diversity in practice, this part of the content must rely on accurate data for more investigation. Another area for improvement is the automatic generation of semantic structure, which is not discussed in detail in this paper, mainly because it is beyond the scope of this paper. How-

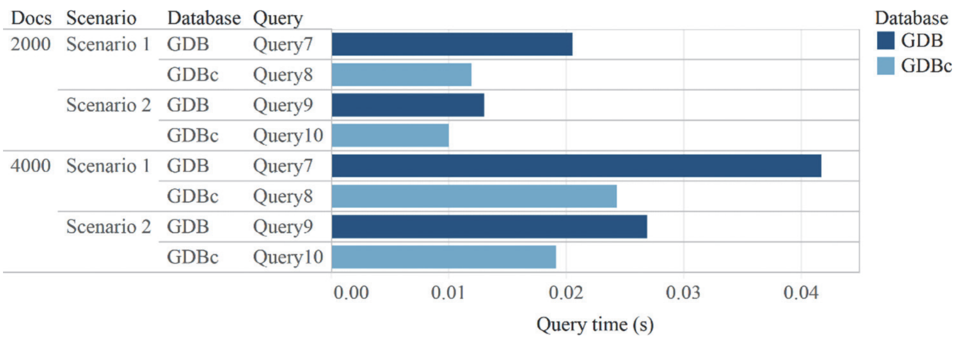

Figure 12: Response time with different storage sizes.

ever, there is no doubt that the generation of semantic structure must be automated, and more natural language processing methods must be studied.

Future work will prioritize expanding the framework's scope in the CBP decision-making process, particularly in project portfolio generation and cross-project association based on the knowledge base. Furthermore, comprehensive testing is required to improve the identification of capability semantic expressions. We intend to use more deep learning-based Natural Language Processing methods to improve this process's automation.

References

Aldea, Adina, Maria Eugenia Iacob, Jos Van Hillegersberg, Dick Quartel, and Henry Franken. 2015. "Capability-based Planning with ArchiMate-Linking Motivation to Implementation." *Proceedings of the 7th International Conference on Enterprise Information Systems*, ICEIS, vol. 2. SciTePress. 352-359. https://doi.org/10.5220/0005468103520359.

Bagchi, Mayuch. 2021. "Towards Knowledge Organization Ecosystem (KOE)." *Cataloging & Classification Quarterly* 59, no. 2: 740–56. https://doi.org/10.1080/01639374.2021.1998282

Benson, Austin R., David F. Gleich, and Jure Leskovec. 2016. "Higher-order organization of complex networks." *Science* 353, no. 6295: 163-66. https://doi.org/10.1126/science.aad9029

Bizer, Christian, Tom Heath and Tim Berners-Lee. 2011. "Linked Data: The Story so Far." In *Semantic Services, Interoperability and Web Applications: Emerging Concepts*, edited by Amit Shet, 205-227. Hershey, PA: Information Science Reference, 205227.

Brown, David. 2000. "Enterprise Architecture for DoD Acquisition." *Acquisition Review Quarterly* 7: 121–130.

Davis, Paul K. 2002. *Analytic Architecture for Capabilities-Based Planning, Mission-System Analysis, and Transformation*." Santa Monica: Rand Corporation.

De Spiegeleire, Stephan. 2011. "Ten Trends in Capability Planning for Defence and Security." *RUSI Journal* 156, no. 5: 20-8.

Dibowski, Henrik, Stefan Schmid, Yulia Svetashova, Cory Henson, and Tuan Tran. 2020. "Using Semantic Technologies to Manage a Data Lake: Data Catalog, Provenance and Access Control." In *13th International Workshop on Scalable Semantic Web Knowledge base Systems*, Athens Greece.

Drury, Brett, Robson Fernandes, Maria Fernanda Moura, and Alneu de Andrade Lopes. 2019. "A Survey of Semantic Web Technology for Agriculture." *Information Processing in Agriculture* 6, no. 4: 487-501. https://doi.org/10.1016/j.inpa.2019.02.001

Gerber, Aurona, Paula Kotzé, and Alta Van der Merwe. 2010. "Towards the Formalisation of the TOGAF Content Metamodel Using Ontologies." *Proceedings of the 12th International Conference on Enterprise Information Systems, Volume 2, AIDSS, Funchal, Madeira, Portugal, June 8 - 12, 2010,* edited by Joaquim Filipe and José Cordeiro. SciTePress.

Ghosh, Shiv Shakti, Subhashis Das, and Sunil Kumar Chatterjee. 2020. "Human-centric Faceted Approach for Ontology Construction." *Knowledge Organization* 47, no. 1: 31–44. https:// doi.org/10.5771/0943-7444-2020-1-31

Hinkelmann Knut, Aurora Gerber, Dimitris Karagiannis, Barbara Thoenssen, Alta Van der Merwe, and Robert Woitsch. 2016. "A New Paradigm for The Continuous Alignment Of Business And IT: Combining Enterprise Architecture Modelling And Enterprise Ontology." *Computers in Industry* 79: 77-86. https://doi.org/10.1016/j.compind.2015.07.009

Hjørland, Birger. 2007. "Semantics and Knowledge Organization." *Annual Review of Information Science and Technology* 41, no.1: 367-405. https://doi.org/10.1002/aris.2007.1440410115

Horrocks, Ian, et al. 2004. "SWRL: A Semantic Web Rule Language Combining OWL and RuleML." *W3C Member Submission* 21, no. 79: 1-31.

House, White. 2007. "FEA Consolidated Reference Model Document Version 2.3." Executive Office of the President of the United States.

Hoyland, Christine A. 2012. "RQ-Tech, a Strategic-Level Approach For Conceptualizing Enterprise Architectures." *Procedia Computer Science* 12: 37-42. https://doi.org/10.1016/j.procs.2012.09.026

Hoyland, Christine A., Kevin MacG. Adams, Andreas Tolk, and Li D. Xu. 2014. "The RQ-Tech Methodology: A New Paradigm for Conceptualizing Strategic Enterprise Architectures." *Journal of Management Analytics* 1, no.1: 55-77. https://doi.org/10.1080/23270012.2014.889912

Jensen, Jesper. 2019. "A Systematic Literature Review of The Use of Semantic Web Technologies In Formal Education." *British Journal of Educational Technology* 50, no. 2: 505-17. https://doi.org/10.1111/bjet.12570

Kebede, Rahel, Annika Moscati, He Tan, and Peter Johansson. 2022. "Integration of Manufacturers' Product Data in BIM Platforms Using Semantic Web Technologies." *Automation in Construction* 144: 104630. https://doi.org/10.1016/j.autcon.2022.104630

Knublauch, Holger, Ray W. Fergerson, Natasha Noy, and Mark Alan Musen. 2004. "The Protégé OWL Plugin: An Open Development Environment for Semantic Web Applications." *Proceedings of The Semantic Web–ISWC 2004: Third International Semantic Web Conference, Hiroshima, Japan, November* 7-11.Springer Berlin Heidelberg.

Kossakoski, Peter 2005. "Capabilities-Based Planning: A Methodology for Deciphering Commander's Intent." *10th International Command and Control Research and Technology Symposium*, Washington.

Lee, Joongyoon, and Youngwon Park. 2009. "A Study on The Abstracted Metamodel Of Dodaf 2.0 For CBA Methodology Execution." *10th ACIS International Conference on Software Engineering, Artificial Intelligences, Networking and Parallel/Distributed Computing*, 27-29 May 2009, Daegu, Korea (South). IEEE. https://doi.org/ 10.1109/SNPD.2009.101.

Lo, Edward HS, Nicholas Tay, and Gary J. Bulluss. 2020. "Program Viewer-A Defence Portfolio Capability Management System." *International Journal of Intelligent Defence Support Systems* 6, no.1: 1-20. https://doi.org/10.1504/IJIDSS.2020.109157

Martin, James N. 2022. "Extending UAF for Model-Based Capability Planning and Enterprise Portfolio Management." *INCOSE International Symposium* 32, no. 1: 15-35. https://doi.org/10.1002/iis2.12915

McGuinness, Deborah L., and Frank Van Harmelen. 2004. "OWL Web Ontology Language Overview." *W3C Recommendation* 10.10: 2004.

Miao, Hong, et al. 2020. "Integrating Technology-Relationship-Technology Semantic Analysis And Technology Roadmapping Method: A Case Of Elderly Smart Wear Technology." *IEEE Transactions on Engineering Management* 69, no. 1: 262-78.

Narayanasamy, Senthil Kumar, Kathiravan Srinivasan, Yuh-Chung Hu, Satishkumar Masilamani, and Kuo-Yi Huang. 2022. "A Contemporary Review on Utilizing Semantic Web Technologies in Healthcare, Virtual Communities, and Ontology-Based Information Processing Systems". *Electronics* 11, no.3: 453. https://doi.org/10.3390/electronics11030453

Padmavathi, Thimmaiah, and Madaiah Krishnamurthy 2017. "Semantic Web Tools and Techniques For Knowledge Organization: An Overview." *Knowledge Organization* 44, no. 4: 273–90. https:// doi.org/10.5771/0943-7444-2017-4-273

Papazoglou, Anastasios. 2014. "Capability-based planning with TOGAF® and ArchiMate®." Master's thesis, University of Twente.

Roach, Terry. 2011. "CAPSICUM-a Semantic Framework for Strategically Aligned Business Architecture". PhD diss., UNSW Sydney. https://doi.org/10.26190/unsworks/15182

Smiraglia, Richard P. 2015. *Domain Analysis for Knowledge Organization: Tools for Ontology Extraction*. Waltham, MA: Chandos.

Tang, Xiao-Bo, Guan-Chao Liu, Jing Yang, and Wei Wei. 2018. "Knowledge-based Financial Statement Fraud Detection System: Based on an Ontology and a Decision Tree." *Knowledge Organization* 45, no. 3: 205–19. https:// doi.org/10.5771/0943-7444-2018-3-205

Torkjazi, Mohammadreza, Arturo J. Davila-Andino, Ahmad Alghamdi, and Abbas K. Zaidi. 2022. "UAF Strategic Planning for Enterprises." *IEEE Access* 10: 123549-123559.

Urbaczewski, Lise, and Stevan Mrdalj. 2006. "A Comparison of Enterprise Architecture Frameworks." *Issues in Information Systems* 7, no. 2: 18-23. https://doi.org/10.48009/2_iis_2006_18-23

8.0 Appendix: SPARQL query test set

The test set of SPARQL queries was proposed to assess the CBP application potential based on the real scenario. Query 1 to Query 6 realize the semantic compliance checking function in Table 4 and modify the triple data through SPARQL's update interface. Query 7 to Query 9 use the knowledge base that has been processed by semantic rules compliance to query and link capabilities to their solutions and motivations.

Prefixes
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> PREFIX owl: <http://www.w3.org/2002/07/owl#> PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> PREFIX xsd: <http://www.w3.org/2001/XMLSchema#> PREFIX cbp: <http://www.ex.org/cbp#>

Query 1. Purpose: *CS developedBy Pr enable C→Pr develop CS enable C*

```
INSERT {?cs cbp:enable ?cd}
WHERE {
 ?cs cbp:developedBy ?project.
 ?project cbp:enable ?cd.
 ?project rdf:type cbp:Project
}
```

Query 2. Purpose: *CS constrainedBy E enable C→CS enable CS, and CS constrainedBy E*

```
DELETE {?environment cbp:enable ?cd}
INSERT {?cs cbp:enable ?cd}
WHERE {
 ?cs cbp:constrainedBy ?environment.
 ?environment cbp:enable ?cd.
 ?cs rdf:type cbp:CS.
 ?environment rdf:type cbp:Environment.
 ?cd rdf:type cbp:CD
}
```

Query 3. Purpose: *CM constrainedBy E enhance Pf→CM constrainedBy E, and CM enhance Pf*

```
DELETE {?environment cbp:enhance ?performance}
INSERT {?cm cbp:enhance ?performance}
WHERE {
 ?cm cbp:constrainedBy ?environment.
 ?environment cbp:enhance ?performance.
 ?cm rdf:type cbp:CM.
 ?environment rdf:type cbp:Environment.
 ?performance rdf:type cbp:Performance
}
```

Query 4. Purpose: *CM enhance Pf support F→CM enhance Pf, and CM support F*

```
DELETE {?performance cbp:support ?function}
INSERT {?cm cbp:support ?function}
WHERE {
 ?cm cbp:enhance ?performance.
 ?performance cbp:support ?function.
 ?cm rdf:type cbp:CM.
 ?performance rdf:type cbp:Performance.
 ?function rdf:type cbp:Function
}
```

Query 5. Purpose: *CS support M associate F*→add *CS support F*

```
INSERT {?cs cbp:support ?function}
WHERE {
 ?cs cbp:support ?missionneed.
 ?missionneed cbp:associate ?function.
 ?cs rdf:type cbp:CS.
 ?function rdf:type cbp:Function.
 ?missionneed rdf:type cbp:MissionNeed
}
```

Query 6. Purpose: *CS support F enable CD→CS enable CD support F*

```
DELETE {?function cbp:enable ?cd}
INSERT {?cd cbp:support ?function.
      ?cs cbp:enable ?cd}
WHERE {
 ?cs cbp:support ?function.
 ?function cbp:enable ?cd.
 ?cs rdf:type cbp:CS.
 ?function rdf:type cbp:Function.
 ?cd rdf:type cbp:CD
}
```

Query 7.

```
SELECT (COUNT(?x) AS ?count)
WHERE {
 {
  ?x rdf:type cbp:Project.
  ?y rdf:type cbp:Performance.
  ?x cbp:develop/cbp:support/cbp:enhance ?y
 }
 UNION
 {
  ?x rdf:type cbp:Project.
  ?y rdf:type cbp:Performance.
  ?x cbp:develop/cbp:support/cbp:constrainedBy/cbp:enhance ?y
 }
}
```

Query 8.

```
SELECT (COUNT(?x) AS ?count)
WHERE {
 ?x rdf:type cbp:Project.
 ?y rdf:type cbp:Performance.
 ?x cbp:develop/cbp:support/cbp:enhance ?y
}
```

Query 9.

```
SELECT (COUNT(?x) AS ?count)
WHERE {
 {
  ?project rdf:type cbp:Project.
  ?project cbp:develop ?z.
  ?z cbp:enable ?x.
  ?z cbp:include ?y.
 }
 UNION
 {
  ?project rdf:type cbp:Project.
  ?project cbp:develop ?z.
  ?project cbp:enable ?x.
  ?z cbp:include ?y.
 }
}
```

Query 10.

```
SELECT (COUNT(?x) AS ?count)
WHERE {
 ?project rdf:type cbp:Project.
 ?project cbp:develop ?z.
 ?z cbp:enable ?x.
 ?z cbp:include ?y.
}
```