applied sciences

MDPI

Article

Building a Design-Rationale-Centric Knowledge Network to Realize the Internalization of Explicit Knowledge

Gaofeng Yue [1,2], Jihong Liu [1,*], Qiang Zhang [1] and Yongzhu Hou [3]

[1] School of Mechanical Engineering and Automation, Beihang University, Beijing 100191, China
[2] China National Institute of Standardization, Beijing 100191, China
[3] Beijing Institute of Mechanical and Electrical Engineering, Beijing 100072, China
* Correspondence: ryukeiko@buaa.edu.cn

Abstract: A large number of publicly available documents, including patent documents and journal articles, can provide designers with creative stimuli, which could facilitate product innovation and collaborative design. As an important tacit knowledge, the acquisition, sharing, and reuse of design rationale (DR) is of great value to designers, which could help designers to better understand design intentions and ideas, support design automation, and promote better collaborative design. However, due to the fragmentation of DR in documentation, this hinders designer acquisition and reuse. If the DR fragments could be automatically extracted from the technical documents to build an interconnected knowledge network system, the problem would be effectively solved, which would further promote the development and utilization of digital archives. To address this issue, this study proposes a three-dimensional design knowledge network metamodel, Design Knowledge Semantic Network (DKSN), and a DKSN-based knowledge fusion method for the construction of a Design Knowledge Network (DKN). We set up an empirical experiment to verify the feasibility and performance of the method. Patent documents and open access research articles are used as sample documents, and a product data dictionary imported from ISO/TS 23768-1 is used as the predefined artifact dictionary. The results further confirm the feasibility and good application prospects of the proposed method.

Keywords: Design Knowledge Network; Design Rationale; knowledge fusion

check for updates

Citation: Yue, G.; Liu, J.; Zhang, Q.; Hou, Y. Building a Design-Rationale-Centric Knowledge Network to Realize the Internalization of Explicit Knowledge. *Appl. Sci.* **2023**, *13*, 1539. https://doi.org/10.3390/app13031539

Academic Editor: Luis Javier Garcia Villalba

Received: 29 November 2022
Revised: 8 January 2023
Accepted: 13 January 2023
Published: 25 January 2023

Copyright: © 2023 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 (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the advent of the era of the knowledge economy, knowledge and innovation play an increasingly important role in the survival and development of enterprises. All innovation, bar none, is designed. No innovation could be brought to market without going through at least one design phase [1]. New value chains are formed as new strategic models by merging diverse technologies to bring about new products, services, and business models that can give a company new competitiveness [2]. Innovative design is a typical knowledge-intensive activity. Designers need to retrieve a large amount of existing design knowledge and information, find other solutions that can be used for reference, and provide inspirational stimuli, ideas, or decision support [3–8]. The novelty of each candidate design is primarily dependent on the process of inspiration and on the way in which information is integrated during the generation of new solutions [4]. Sources of inspiration can help designers to define context, to trigger ideas, and to build a designer's mental representation of their design during the design thinking process, which is becoming a central point for the industry to seek innovative solutions to problems [4,9,10]. Effective design knowledge retrieval is essential to improving the creativity of designers [3,11,12], and knowledge networks help designers to quickly find and understand knowledge from unstructured technical documents.

DR explains how and why products are designed that way [13], which has an important role in helping designers to better understand design intentions and ideas, to support

design automation [14] and so to promote better design collaboration [15–19]. DR is essentially a kind of tacit design knowledge [20,21], which also means that designers cannot or are unwilling to record it completely [22]. If the designer records the rationale knowledge during the design process, it may disturb the designer's thinking process [23–25]. On the other hand, knowledge is often treated by designers as their own asset, and they are unwilling to actively share it. However, technical literature such as patents and academic journals may more or less contain some relevant DR. Meanwhile, this also leads to the fragmentation of rationale knowledge [26].

Researchers have tried to solve the DR fragmentation problem through document integration. In the late 1980s, Jeff and Michael [27] proposed an application-oriented hypertext system based on IBIS, gIBIS, which promotes the integration of fragmented rationales through color and relational databases to support collaboration among team members. Based on the IBIS model, Rob et al. [3] proposed a tool called Design Rationale editor (DRed), which utilizes a directed graph of dependencies to capture issues, options, and pros and cons in the design thinking process, and to build a DKN. Wang et al. [28] argued that design knowledge gained from various sources requires integrated representations to effectively and efficiently reuse pieces of DR, and proposed a DRed-based integrated knowledge representation and corresponding DR retrieval framework. To achieve synergy in product design and manufacturing tool design under mass customization, Morteza et al. [17] proposed a method to enable the acquisition and representation of DR in different formats such as CAD models, spreadsheets, textual format, and web pages.

The complexity of technical documentation, involving different writing styles, multiple topics and purposes, and language expression habits and professional backgrounds, greatly hinders the designer's reading comprehension of engineering knowledge. [29,30]. The file-based methods can no longer meet the requirements of the digital, networked, and intelligent era, and researchers have begun to study the automatic acquisition and formal representation of rationale knowledge. B. Rogers et al. [31] and M. Lester et al. [32] conducted a study of automatic DR text extraction using the bug reports of Chrome Browser as the research object. Z. Kurtanović et al. [33] extracted user rationale about how to select software application decisions from 32,414 online reviews of 52 software applications on Amazon, using multiple classifier algorithms. To address the problem that the existing DR models largely rely on human efforts in capturing DRs along design activities, Y. Liu et al. [34] proposed an ISAL (Issue, Solution, Artifact Layer) model for rationale information extraction from the design archival documents. On the basis of the ISAL model, Y. Zhang et al. [35] proposed the ISAA model, classifying the DR information into four classes, issues, solutions, arguments, and artifacts, which can be utilized to construct a DKN. However, the ISAA model does not consider how DR is automatically extracted from documents, and the ISAL model does not support knowledge networks.

Researchers have conducted numerous studies on the integration, sharing, and rapid retrieval of design knowledge. However, even so, how to build a DKN for knowledge in one document, and how to build an Open Design Knowledge Network (ODKN) for knowledge in multiple documents, which can be automatically interpreted by computers and can facilitate quick understanding by designers, thus providing designers with creative and inspirational stimuli, is still a problem. To this end, this paper proposes a three-dimensional knowledge network metamodel, including the literature, the artifact, and the rationale. The metamodel can be used for the knowledge extraction of technical documents, and the construction of Knowledge Graphs (KG), thereby facilitating the discovery, understanding, transfer, and reuse of knowledge.

The main contributions of this paper can be summarized as follows:

- The DKSN, a knowledge network metamodel based on DR theory, is proposed to provide designers with innovative ideas, inspirations, and creative stimuli.
- The DKSN takes into consideration the requirements of text mining, making it a convenient, rapid, and efficient approach for the capture, sharing, and dissemination of design knowledge in documents.

- It can be used to integrate DR from patents, scholarly articles, and other publicly available documents to build an ODKN.

The rest of this paper is organized as follows: Section 2 reviews the relevant work on DR representations, design knowledge networks, and semantic network and knowledge fusion. Section 3 presents a three-dimensional knowledge network metamodel, and a knowledge fusion method. Section 4 sets up an illustrative example. Finally, the conclusions are discussed in Section 5, together with potential future work.

2. Related Work

2.1. Design Rationale Representations

Horst and Melvin [36] believe that scientific issues are "tame" problems, but that problems of planning cannot be meaningfully correct or false, and are "wicked" problems. On the basis of this theory, an issue-based information system (IBIS) model was proposed to address the co-ordination and planning of political decision-making processes [37]. The design is seen as a process of argumentation instead of a sequence of understanding the problem, analysis, synthesis, and implementation. Based on the IBIS model, gIBIS [27], DRL ontology [26], QOC [38], PHI, and Kuaba ontology [39] were proposed as DR representations. The DRed was developed as a DR tool for designers to record their rationale as the design proceeds. Issues, options, and associated pro and con arguments could be captured by the DRed in the form of a directed graph of dependencies [3]. These argumentation-based DR models are mostly used in the early conceptual design stage. Liu et al. [34] proposed an ISAL model for DR extraction from technical documents, and Yingzhong et al. [35] proposed an ISAA model to build a semantic design knowledge network.

Hiroyuki [22] considers the design system as a knowledge-intensive system, and put forward the concept of a design knowledge system. Hiroyuki classified design knowledge into tacit knowledge and codified knowledge. Tacit knowledge resides in a human brain, which is not easily codified or captured. Furthermore, design knowledge can also be divided into design process knowledge and design object knowledge. Design process knowledge describes how, whereas design object knowledge describes what. Design is a cognitive process of designers, which is a stepwise transformation process from the function space to the attribute space [40]. On this basis, Tetsuo [40] put forward the concept of design-intent modeling to record the design decision processes. Liu and Hu [41] proposed an intention-driven DR evolution model to model the cognitive design process. To achieve design collaboration, Cheng and Ma [42] proposed a functional feature-based CAD modeling method, and Cheng et al. [43] proposed a feature semantic network for communicating design intent between heterogeneous CAD systems.

Table 1 is a brief list of the DR representations.

Table 1. DR representations list.

Literature	DR Representation	Application Field
[37]	IBIS: issue, position, argument (support or oppose)	political decision processes; general design
[44]	DRL: decision problem, alternative, goal, argument, decision, viewpoint, question, DRL relations	general design
[38]	QOC: questions, options, and criteria	human–computer interaction
[39]	Kuaba: question, idea, reasoning element, artifact, decision, argument, justification	model-based designs, software design
[41]	intent, option, justification, decision, operation	engineering design

Table 1. *Cont.*

Literature	DR Representation	Application Field
[43]	design intent: artifact, design history, participants, artefact structure, boundary representation	2D CAD
[34]	ISAL: issue, solution, artifact	engineering design
[35]	ISAA: issue, solution, argument, artifact and evidence	engineering design
[32]	decision, alternative, answer, argument, assumption, decision, procedure, question, requirement	bug reports for software
[33]	issue, alternative, criteria, decision, justification	user online reviews for software application
[42]	part, assembly, functional requirement, geometry feature, constraint	CAD model
[20]	issue, solution (usage, performance, accuracy, product, material, structure, function)	engineering design

These studies gradually formed the DR theory: DR is strategic tacit knowledge, and in different application scenarios, there are some common constituent elements of DR. These studies are of great value to knowledge management. However, there is still a gap between the DR knowledge contained in the massive technical documentation and how to achieve the acquisition, sharing, and understanding of DR knowledge. Most of these studies focused on DR capture during the design process. Only the ISAL model is a DR representation for engineering design knowledge extraction from the documents. However, the most important information is not included in the ISAL model, such as the *pros* and the *cons*. In addition, the extracted DR is mostly paragraphs which are not intuitive for designers as inspirational stimuli.

2.2. Design Knowledge Network

Knowledge network in cognitive science. Based on mathematical graph theory [45], networks provide simple and powerful representations of complex systems comprising nodes and edges, which have long played an important role in cognitive science. Studies have shown that semantic networks of highly creative individuals have higher connectivity and shorter distances between concepts, compared to low-creative individuals [45,46]. Liu et al. [47] regard the designer's tacit knowledge as semantic memory, which can be represented through a semantic network. This semantic network is constructed as conceptual nodes through lexical items, phrases, or information that most designers can easily understand. On this basis, Liu et al. [47] proposed a function–structure concept network (FSCN) to build associations between design information, so as to inspire idea generation in the conceptual design. George Siemens and Stephen Downs proposed a connectivistic view of knowledge [48,49] and hold that: learning is primarily a network forming process; knowledge rests in networks; and the ability to see connections and to recognize patterns and make sense between knowledge is the core skill for individuals. Based on this theory, Johansson et al. [49] proposed a method to establish an infrastructure to meet design automation for mass customization, through the use of graph theory, information filtering, and the quality assurance of CAD models. Sarica and Luo [50] propose a technology semantic network to help engineers to quickly understand complex technical design knowledge. Loureiro et al. [51] proposed a design structure network with design variables as nodes and interdependence relations as edges for the visual representation of product design specifications, so as to facilitate communication between specialists and non-specialists.

Knowledge networks for the knowledge management of technical documentation. Addressing the problem that complex product development increasingly relies on design collaboration and knowledge sharing, Peng et al. [12] proposed a knowledge hypernetwork consisting of a designer network, a product network, a problem network, and a knowledge unit network, to explicitly capture the relationships between various elements in a multifaceted knowledge representation. To help designers to quickly find knowledge from numerous unstructured text documents, Hao et al. [29] proposed a decision-making knowledge network metamodel from three dimensions, including domain knowledge, people knowledge, and decision knowledge, for integrating and navigating knowledge related to engineering design decisions. Chen and Luo [52] proposed a knowledge network approach to facilitate knowledge exploration in literature abstracts that included six aspects: authors, affiliations, background, objective, solution, and findings.

Knowledge network based on citation relationships. Citation network analysis has provided a macro perspective by using large datasets to help solve research, management, or information service issues, such as university rankings, research evaluation, or the visualization of knowledge areas [53]. The titles, keywords, abstracts, and reference relationships in academic literature are used to establish and to evaluate networks to track the evolution of scholarly knowledge. Based on a co-word network, Li et al. [54] adopts semantic similarity calculation and fuzzy neural network prediction for constructing empirical engineering knowledge (EEK) networks. With the rapid growth of digital libraries, the large amount of scholarly data pose increasing challenges for researchers and publishers to analyze scholarly information. M. et al. [55] proposed a graph-based approach consisting of nodes and edges to deal with the big data problems of scholarly literature, including selecting the right reviewers for submitted papers, finding quality impact factors, and ranking journals and researchers.

The design knowledge network provides an effective technical means for the acquisition, rapid retrieval, and understanding of design knowledge. However, it is still an issue regarding how to build a DKN to realize the acquisition, storage, and sharing of DR knowledge in technical documents.

2.3. Semantic Network and Knowledge Fusion

As a formal knowledge representation, ontology plays an important role in knowledge acquisition, storage, sharing, and reuse. Thomas and Daniel [13] proposed an ontology of design knowledge for DR capturing. Following this, several studies on DR ontology were carried out [26,35,39,44]. Semantic Web Rule Language (SWRL) is an expressive OWL-based rule language, which provides deductive reasoning capabilities. W3C's Semantic Web technology promotes the application of ontology in knowledge representation and acquisition. The knowledge graph was originally used by Google to improve search engine performance [56]. Hogan et al. [57] define the knowledge graph as a graph of data intended to accumulate and convey knowledge of the real world, whose nodes represent entities of interest, and whose edges represent relations between these entities. The semantic network and knowledge graph provide the basic method for ontology turtle modeling, which can be utilized to construct a formalized knowledge network. Knowledge fusion is an effective solution to the problem of acquiring knowledge from different sources [58]. Addressing the multi-dimensional, heterogeneous, and time series issues on the data collection of the manufacturing process, Liu et al. [59] use the knowledge fusion process to disambiguate, integrate, and reason the knowledge, and to obtain high-quality knowledge after eliminating a series of redundant and erroneous information. The knowledge fusion process includes the entity alignment or cognition matching of entities, relations, and attributes, which is one of the fundamental techniques for building knowledge graphs [59–61]. Ji et al. [62] proposed a BERT-based semantic similarity calculation method to realize the fusion of entities and relations in the knowledge graph.

These ontology-based semantic webs, knowledge graphs, and knowledge fusion technologies are the basis for the DKN construction, which also provides operable methods for the improvement of knowledge networks.

3. Proposed Method

3.1. Framework of the Proposed Method

To realize the acquisition, storage, and understanding of the design knowledge contained in technical documents, the concepts of DKN and ODKN are presented. Based on the semantic web, ontology, and knowledge graph, a metamodel of Design Knowledge Semantic Network (DKSN) is proposed to construct the knowledge network. The DKSN covers the ontologies of three dimensions, including the *literature* ontology, the *artifact* ontology, and the *rationale* ontology, and relations that connect the knowledge of the three dimensions to construct an organic knowledge network system. As shown in Figure 1, there are three steps in the construction of a design knowledge network:

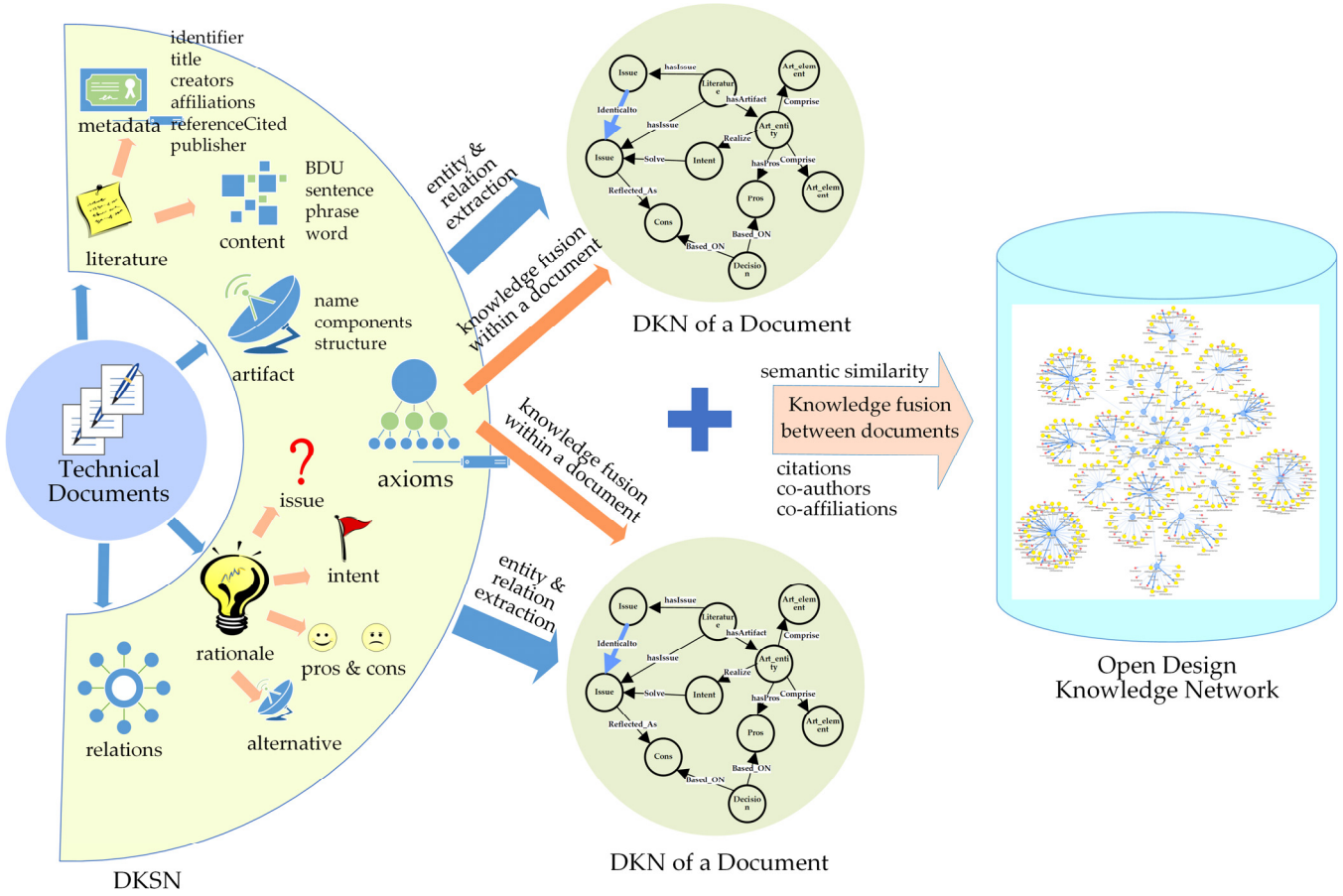


Figure 1. Framework of the study.

According to the ontology definition in DKSN, relevant knowledge entities and associated relationships are extracted from a technical document to build the DKN of a document.

According to the axioms, the entities and relationships inside the document could be aligned and merged, eliminating redundant knowledge within a document.

To establish a broader ODKN, knowledge fusion is achieved from two perspectives: the content perspective, using semantic similarity analysis, and the document metadata perspective, using citations, co-authors, co-affiliation, and other document metadata.

3.2. Design Knowledge Network

The SECI Model (Socialization, Externalization, Combination, Internalization) was proposed by Nonaka and Takeuchi [63] for the management of organizational knowledge. Nonaka et al. [64] believe that the SECI model is a process model, which is about knowledge sharing and the knowledge conversion process, and that new knowledge is created from the continuous interaction of tacit and explicit knowledge. Based on the SECI model, Dubberly and Evenson [65] regard design as a learning process that could be analyzed with the SECI model. Active engagement with the online knowledge network can facilitate a significant harnessing of the intellectual capital [66].

To promote the sharing and reuse of the design knowledge, Jia et al. [20] proposed an approach to capturing tacit design knowledge using knowledge graphs. On this basis, we propose the concept of ODKN, which is a knowledge network composed of knowledge from various documents, which integrates the knowledge of various design teams. As shown in Figure 2, most of the design knowledge is tacit knowledge, especially DR, which usually resides in the designer's mind. Due to the nature of tacit knowledge, it is hard to codify, and only a small number of knowledge pieces are explicitly recorded in patents, design specifications, journal articles, and others. This has also led to the fragmentation of DR. Only by integrating fragmented DR and by building a knowledge network can designers quickly query and discover, and directly understand, to achieve the purpose of design knowledge reuse.

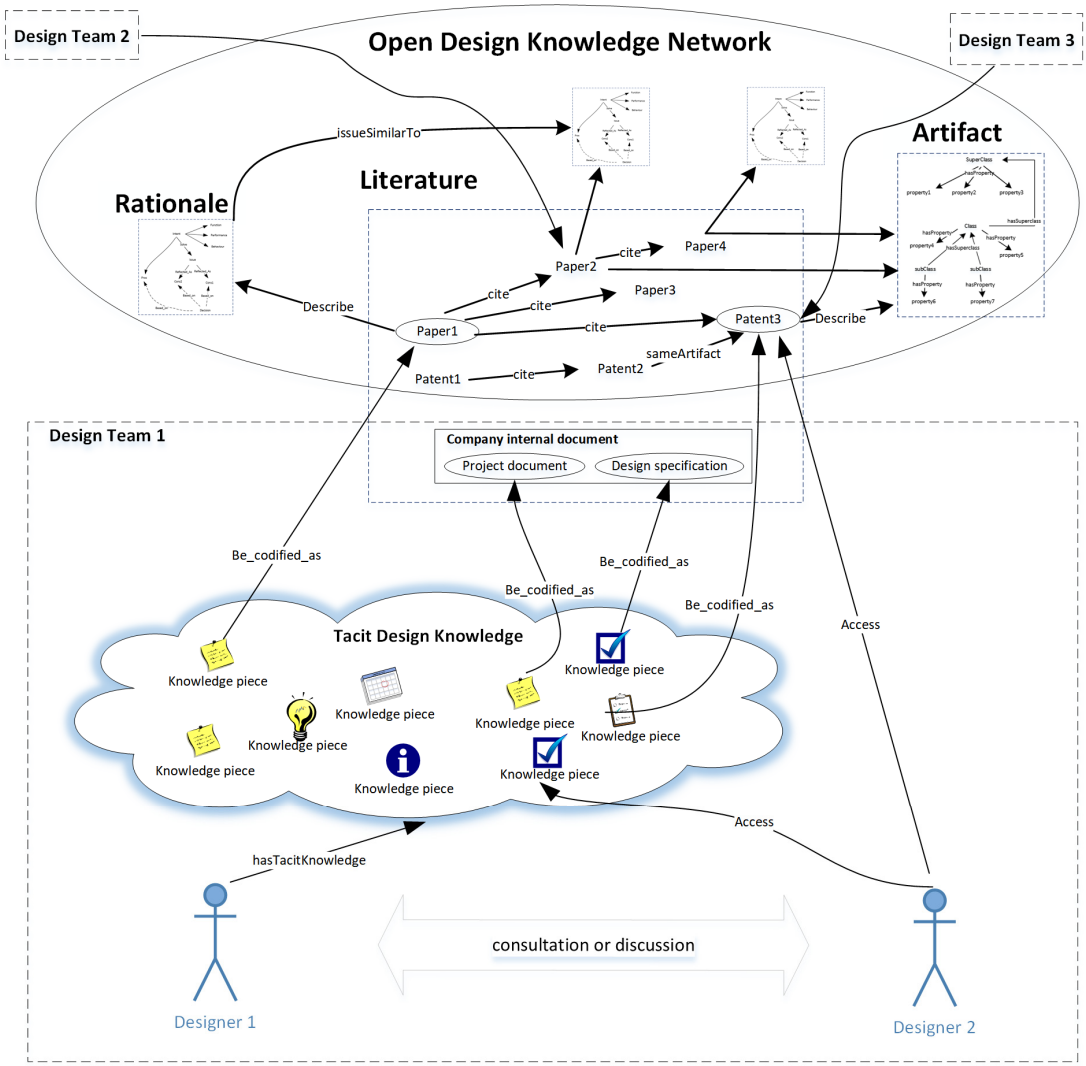


Figure 2. The reference model of tacit design knowledge sharing.

According to the scope of design knowledge sharing, DKN can be divided into design team knowledge networks, intra-organizational knowledge networks, and open knowledge networks between organizations, consisting of various technical documents. The publicly available documents are accessible to all organizations and technicians, usually with strict drafting rules and a formal document structure, and they are also suitable for knowledge extraction based on natural language processing (NLP).

As a graphical knowledge network, DKSN supports the automated extraction and knowledge fusion of knowledge entities and relationships, which is more conducive to the dynamic management of design knowledge. With the advent of the knowledge economy, the number of articles and patents has increased year by year, which has brought difficulties to the browsing, querying, and understanding of knowledge in literature. For newly published technical literature, DKSN supports the real-time, automated extraction of entities and relationships, and it can be integrated into the existing knowledge network system. It can be used by companies, publishers, and libraries in knowledge management platforms, and it can also be used as a knowledge management tool for researchers.

3.3. Design Knowledge Semantic Network

3.3.1. Overview

The DKSN covers three dimensions of knowledge, including *literature* knowledge (metadata, such as author, affiliations, keywords, title, citations, and contents), *artifact* knowledge (the inheritance and composition relationship of design objects), *rationale* knowledge (*issues*, *intents*, *pros*/*cons*, and *alternatives*), and the relationships between them. *Artifact* is about design object knowledge, and *rationale* is about design process knowledge. Both are described in the *literature* in a specific writing style, as shown in Figure 3.

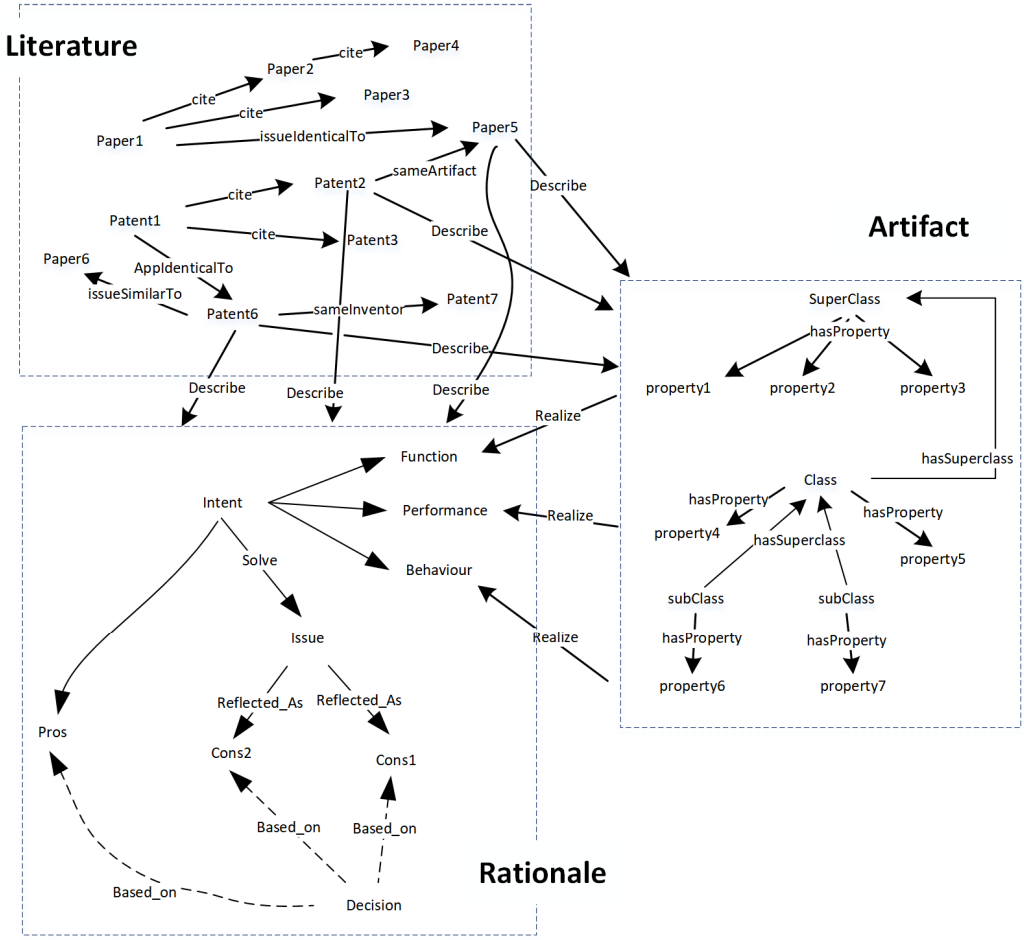


Figure 3. Diagram of Design Knowledge Semantic Network.

Literature dimension. The task of computational linguistics is to transform technical literature that is easy for humans to read into text data that can be easily interpreted by computers. The machine interprets the text data, tries to find and extracts the knowledge therein, and transforms it into a knowledge network that is suitable for the designer's fast understanding. Therefore, this study takes documentation as an important dimension for the construction of DKN. Through citations, co-inventors, co-authors, and others, the publicly available literature naturally constitutes a knowledge network.

Artifact dimension. Simon [67] defined *artifacts* as being "produced by art rather than by nature; not genuine or natural; affected; not pertaining to the essence of the matter". *Artifacts* are the material basis for the realization of design intent. They are about know-what. To put it simply, treating the *artifact* as a product, using the superclass, class, subclass, property, and other elements for ontology modeling, all products in ODKN will form an open, hierarchical product data dictionary, which is also a knowledge network.

Rationale dimension. A *rationale* is a justification of the design decision, including the *issue*, *intent*, *pros* and *cons*, and *alternatives* of the *artifact*. *Alternatives* are other *artifacts*. Design decisions are justified by comprehensively considering the *pros* and *cons* of solutions during the design argumentation. They are about know-why. Due to the fragmentation of rationale, rationale alone cannot form a knowledge network.

Definition 1. *DKSNnetwork. A Design Knowledge Semantic Network is a directed, node-edge-labeled knowledge graph. G = {C, P, R, A}.*

- Classes or Nodes (N). In this paper, entities, nodes, or classes are considered synonyms, which are expressed as "(class)".
- Properties (P). Related information about a class. For example, names, identifiers, authors, and other relevant information can be the property of a document; materials, geometric dimensions, and functional descriptions can be the properties of an artifact.
- Edges (E) or Relations. Edges represent the logical relationships between nodes or classes. They include generic relationships as defined in OWL or protégé tools, such as IS-A; some specific DR relationships within the DKSN network, such as *Realize*, *Describe*, *hasIntent*, and *hasOpionion* relations, are expressed in the form of directed arrows: -[relation]->.
- Axiom rules (A). Axiom Rules are expressed with the Semantic Web Rule Language (SWRL). SWRL is an expressive OWL-based rule language. SWRL allows for the definition of DKSN-specific rules that can be expressed in OWL concepts to provide more powerful deductive reasoning capabilities.

3.3.2. Literature Ontology

Definition 2. *Literature Class. The literature refers specifically to technical documentation in the field of engineering design. Literature has the dual characteristics of linguistics and informatics. From a linguistic viewpoint, literature (including documents, paragraphs, sentences, clauses, and phrases) could be characterized as syntactics, semantics, and pragmatics. Syntactics are usually identified via form, order, format, and structure. Semantics represent the meaning or a relationship of the real world, expressed using phrases or symbols. Pragmatics is about context. The metadata information is the property of the literature, such as the title, creators, affiliations, references cited, publisher, publishing date, and others, as shown in Figure 4.*

Basic Document Unit. Technical literature is a relatively independent and complete knowledge collection, which can contain several independent knowledge units, such as abstract, introduction, background, main body, discussions, summary or conclusion, and others. These parts may maintain a certain independence from the content of the main body, and they can independently form a small document unit, which is called the Basic Document Unit (BDU). The design knowledge in the BDU has an inherent logical structure,

and this structure can be used for knowledge extraction and knowledge fusion. The BDU can be used as a property to indicate the origination of a sentence or a word.

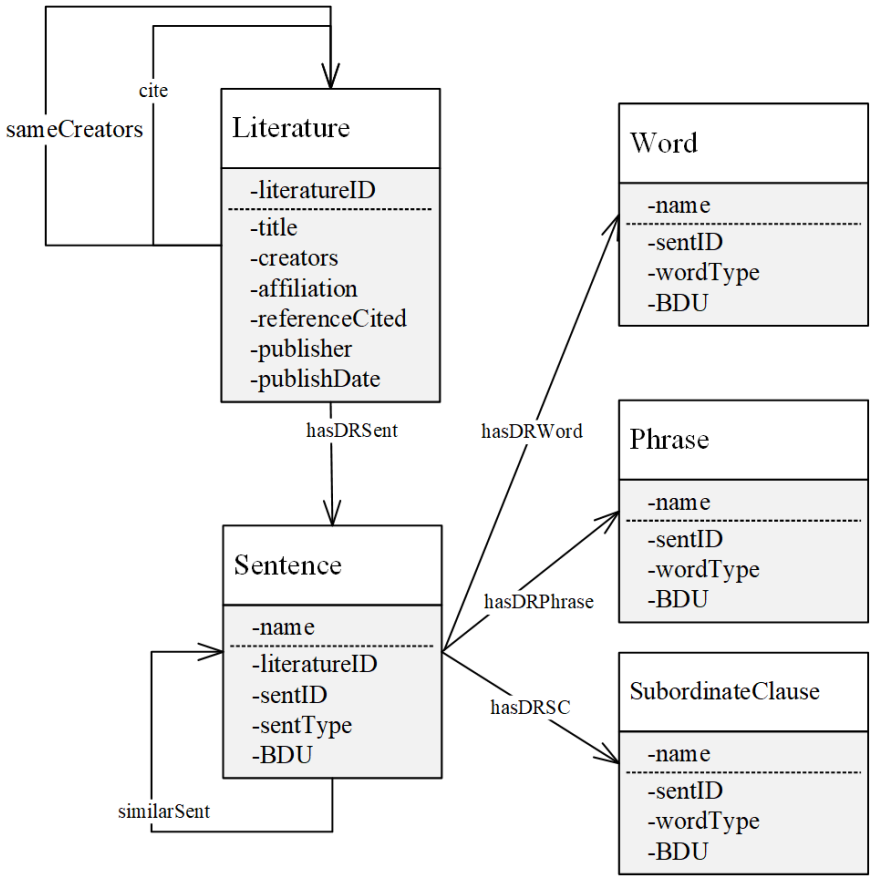


Figure 4. Literature ontology.

Sentence. A sentence is the basic element that makes up the technical documentation. In this study, the sentence is the basic design knowledge unit, such as *issue* sentences, *intent* sentences, *alternative* sentences, *argument* sentences, *artifact* sentences, and others. Sentences are easier for designers to read and to comprehend quickly than paragraphs, although the information that it contains may be incomplete. The attributes of a sentence include the identifier, DR type of the sentence, the content, and the document identifier to which the sentence belongs.

Word, phrase or subordinate clause. Words, phrases, and subordinate clauses form sentences that can express some basic meaning; they are shorter and more intuitive than sentences, and they are easy for designers to understand. Feature words in the technical literature can be divided into the following: stop words, *predefined artifact* words, *literature* words, *positive* or *pro* words, *negative* or *con* words, *issue* words, *alternative* words, *Describe* relation words, *Realize* relation words, *THERE_BE* relation words, and structural relation words, corresponding to the "*wordType*" attribute in Figure 4.

Contain relations. *Contain* represents the aggregation relation within the technical literature. For example, a document *contains* specific sentences, or a sentence *contains* specific words. *Contain* can be embodied as the "*literature hasDRSent* Sentence" pattern, which includes *hasIssue*, *hasIntent*, *hasArtifact*, *hasArgument*, and *hasAlternative*. *Contain* can also be embodied as "*DRSent hasDRWord feature_word*", which includes *hasArgumentWord*, *hasDefArtifact*, *hasAlternative*, and others. *Contain* can be used to represent the *composition* relationship between literature nodes, as shown in Expressions (2) and (3). To express the semantic relationship between entities, this paper adopts the expression method of "node"

and "edge" specified in the Knowledge Graph tool Neo4j. "()" denotes entities or nodes, and "-[]->" denotes relations or edges with directions.

(literature) -[hasDRSent]-> (Sentence) (1)

(Sentence) -[hasDRWord]-> (Word) (2)

Take patent document as the literature—it can include the following sentences:

(patent)-[hasIssue]->(issueSent) (3)

(patent)-[hasIntent]->(intentSent) (4)

(patent)-[hasArgument]->(argumentSent) (5)

(patent)-[hasAlternative]->(alternativeSent) (6)

Furthermore, the sentence can contain some DR words:

(argumentSent)-[hasOpinion]->(position)-[hasArgumentWord]-> (cons or pros) (7)

(intentSent)-[hasDefArtifact]->(preDefinedArtifact) (8)

(intentSent)-[hasArtifactEntity]->(ArtifactEntity) (9)

(ArgumentSent)-[hasArtifactEntity]->(ArtifactEntity) (10)

Cite relations. *Cite* represents citations between technical literature, which indicate a flow of knowledge from the cited to the citing literature [53]. DKNs among documents can be connected through the citation relationship to construct an interconnected knowledge network. In USPTO patents, citations are identified using "References Cited", "Other References", and "CROSS-REFERENCE TO RELATED APPLICATIONS", which can be used to establish the citations between patents. Furthermore, co-authors, affiliations, and other metadata information could also be used to build relations between documents.

3.3.3. Artifact Ontology

Definition 3. *Artifact class. Artifacts, which can also be called design objects, are objects that are created by humans to solve certain problems. Artifacts can be anything designed to achieve a goal, purpose, or function, including machine equipment, devices, computer programs, processes/methods, chemical components, and other man-made objects. An artifact entity is structured by the constituent artifact elements according to a certain structure, as shown in Figure 5.*

Definition 4. *Reference artifact class. To form a knowledge network of artifacts, a reference product data dictionary is introduced. As shown in Figure 5, the reference product data dictionary can be constructed using the ontology modeling approach, including the reference artifact class and the reference property class. These product ontologies consist of hierarchies of characterization classes of products built according to a common methodology that is intended to enable multi-user consistency. The extracted artifacts from documents can be associated with the appropriate reference class or property of the artifacts. Hence, the extracted artifacts can be connected to the knowledge network.*

The following relations could be used for the construction of the artifact network:

(ArtifactEntity)-[Comprise]->(ArtifactElement) (11)

(artifact)-[hasAlternative]->(artifact) (12)

(ArtifactElement)-[beStructured]->(ArtifactElement) (13)

(RefArtifactClass)-[hasSuperclass]->(RefArtifactClass) (14)

(RefArtifactClass)-[hasProperty]->(RefArtifactProperty) (15)

(RefArtifactProperty)-[hasClass]->(RefArtifactClass) (16)

(ArtifactEntity)-[Associateto]->(RefArtifactClass) (17)

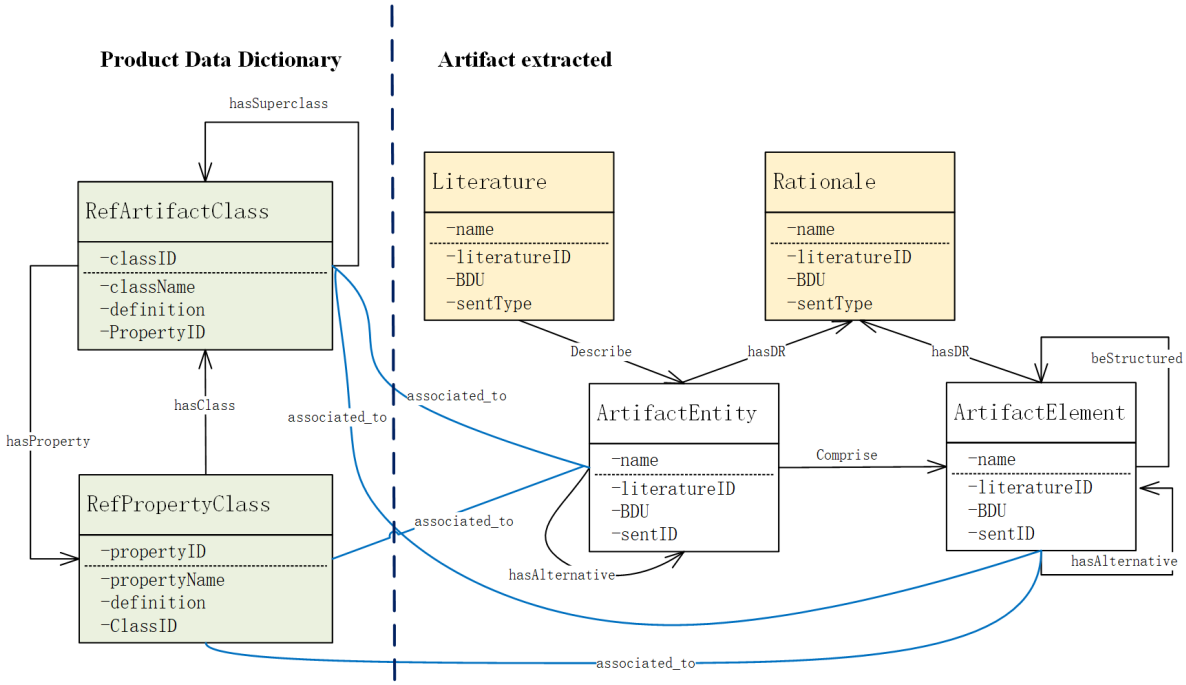


Figure 5. Artifact ontology.

3.3.4. Rationale Ontology

The DR is the core of the DKN. As shown in Figure 6, the model takes into consideration the general writing criteria of technical documents and the structure of the document. At the same time, as inspirational design stimuli for designers, extracted rationale should be as intuitive and short as possible, so that the designer can quickly understand, and there should not be a paragraph or lengthy statement.

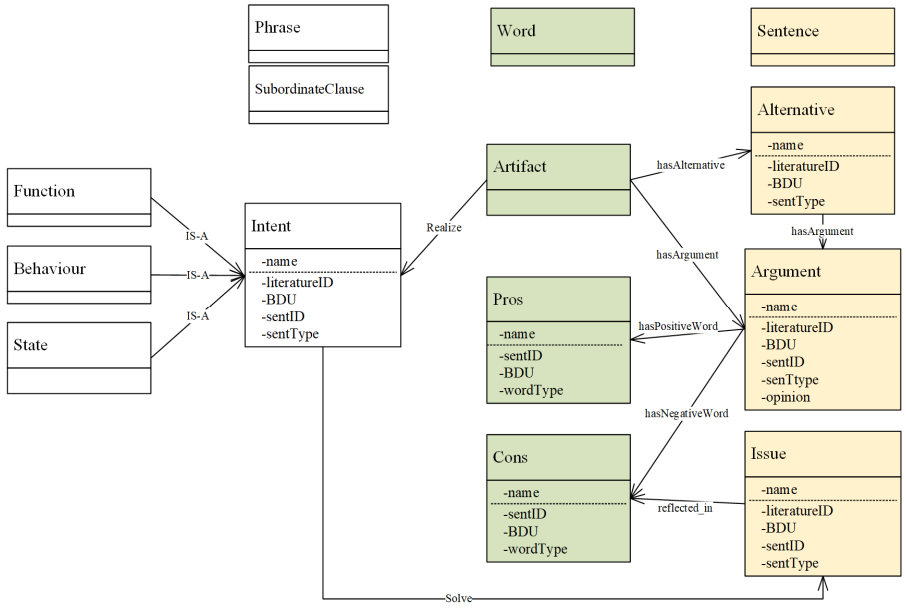


Figure 6. Rationale ontology.

Definition 5. *Rationale Class. Herein, we adopt Lee's definition of DR [68]: DR includes not only the reasons behind a design decision, but also the justification for it, the other alternatives considered, the tradeoffs evaluated, and the argumentation that led to the decision. It includes issues, intents, alternatives, pros and cons of the artifact, and others. Alternatives are other artifacts that address the same design problem. As shown in Figure 6, the rationale information includes the following:*

- *Issue*—problems with existing artifacts or requirements for new artifacts. The detailed description of the *issue* is usually embodied in the description of the shortcomings of existing artifacts, which are generally expressed using negative statements. *Issues* are the core and focus of the entire design document that need to be addressed.
- *Intent*—goals and expectations that the designer wants to achieve with the *artifact*. Design *intent* can include the function of the design object, such as the drone realizing the functions of transportation, maintenance, photography, and others. It can also include the performance *intent* of an artifact, such as safety, economy, reliability, and others; the *artifact* implements a specific behavior or operational *intent*, such as vertical take-off and landing operations of a drone. Hence, design *intent* addresses a design problem or set of problems.
- *Argument*—advantages and disadvantages of the *artifact*, positions for or against, opinion analysis from the designer or interested parties. It can include a. cons, the descriptions of the deficiencies, shortcomings, and negatives of *alternatives* or other traditional *artifacts*; and b. pros, used to express positive information such as excellent functions, reliable performance, and a wide range of applications.
- *Alternative*—other *artifacts*, design options, or relevant solutions.
- Relationships in rationale knowledge networks and other relevant relations include:

(subClass)-[IS_A]->(Class) (18)

(literature)-[Describe]->(artifact) (19)

(artifact)-[Realize]->(intent) (20)

(intent)-[Solve]->(issue) (21)

(argument)-[hasOpinion]->(position) (22)

(artifact)-[hasAlternative]->(alternatives) (23)

(position)–[hasArgumentWord]->(cons or pros) (24)

3.3.5. Axioms

OWL2 Direct Semantics defines a set of axioms, and we employ some general Axioms, such as the *IS-A* relationship. Furthermore, several basic relations for DKSN and some rules are defined. Tables 2 and 3 are some rules that are defined for design knowledge extraction and knowledge fusion.

3.4. Knowledge Fusion

In general, knowledge fusion is used to address the problem of redundancies, ambiguities, and errors in acquiring knowledge from different sources. This study is not aimed at solving the right and wrong problems of knowledge from different sources, but rather at aims to achieve two goals: to merge redundant knowledge from the same document, and

to create an interconnected network to fuse knowledge from different documents. Table 4 lists two types of knowledge fusion tasks, and the approach should be applied accordingly.

Table 2. Sentiment analysis rules.

Literature(?x)^hasArgument(?x,?arg)^hasNegativeWord (?arg,?neg)→hasOpinion(?arg,?neg) Literature(?x)^hasArgument(?x,?arg)^hasPositiveWord (?a,?pos)→hasOpinion(?arg,?pos)	The *positive* words or *pro words* contained in the argument sentence represent a supportive opinion about the *artifact* (design object or solution). Conversely, the *negative* words or *con words* contained in the *argument* sentence of the literature represent an objection opinion to the *alternative artifact*.
Basic_Doc_Unit(?x)^hasIssue(?x,?i)^hasArgumentWord (?i,?neg)→ExpressedAS(?i,?neg)	In the same BDU of the literature, *negative* words are detailed descriptions of the *issue*.

Table 3. Knowledge extraction rules.

(Literature(?lit)^hasIntent(?lit,?i1)^hasArtifact (?i1,?a))^(Literature(?lit)^hasIntent(?lit,?i2)^hasArtifact (?i2,?a))→intentIdenticalTo(?i1,?i2)	If *intent* sentence i1 and *intent* sentence i2 contain the same *artifact elements* and the same design *intent*, the *intent* sentence i1 is identical_to the intent sentence i2.
(Literature(?lit)^hasIssue(?lit,?i1))^(Literature(?lit)^has Issue(?lit,?i2))→issueIdenticalTo(?i1,?i2)	There is only one major *issue* in the same document.
Literature(?lit)^hasMetadata(?lit,?x)^Basic_Doc_ Unit(?y)^hasBDU(?lit,?y) →hasMetadata(?y, ?x)	For the metadata of the literature, they are also the metadata of the BDU of that *literature*.
Literature(?lit)^hasArtifactEntity(?lit,?x)^Comprise (?x,e1)^Literature(?lit)^hasArtifactEntity (?lit,?e2)^swrlb:stringEqualIgnoreCase(?e1,?e2)^same BDU(?e1,?e2)→ artIdenticalTo(?e1,?e2)	In the BDU of the same document, *ArtifactEntity* e1 and *ArtifactElement* e2 are included. If e1 and e2 have equal strings (word or N-gram), then it can be considered that e1 and e2 represent the same *artifact*, so that the *artIdenticalTo* relationship can be established.
Literature(?lit1)^hasIssue(?lit1,?i1)^Literature(?lit2)^has Issue(?lit2,?i2)^swrlb:greaterThanOrEqual(?I_{sim}, 0.8)→ issueIdenticalTo(?i1, ?i2)	*Literature* 1 has *issue* i1, and *literature* 2 has *issue* i2. i1 and i2 have a semantic similarity relationship, and the similarity coefficient $I_{sim} \geq 0.8$, then it can be determined that i1 and i2 represent the same *issue*, so that an *issueIdenticalTo* relationship can be established.
Literature(?lit1)^hasIssue(?lit1,?i1)^Literature(?lit2)^has Issue(?lit2,?i2)^swrlb:greaterThanOrEqual(?I_{sim}, 0.6)^swrlb:lessThan(?I_{sim}, 0.8)→ issueSimilarTo(?i1, ?i2)	*Literature* 1 has *issue* i1, and *literature* 2 has *issue* i2. i1 and i2 have a semantic similarity relationship, and $0.6 \leq I_{sim} < 0.8$. It can be determined that i1 and i2 represent the similar *issue*, so that an *issueSimilarTo* relationship can be established.

Table 4. DR fusion methods.

Scope of Fusion	DR Type	Methods
within a document	issue, intent, alternative, pros and cons	semantic similarity
	artifacts, pros, cons	word comparison
between documents	issue, intent	semantic similarity
	literature	citations, co-authors, co-affiliations, and others

For technical documents such as patents, the same semantic content is often repeated within a document, which could lead to knowledge redundancy. Potential entity fusion in DKN includes *issues* (sentence), *intents* (phrase, clause, or sentence), *alternatives* (sentence), *pros* and *cons* (word), *artifacts* (word), and others. To reduce this redundant knowledge, duplicate entities should be merged. Herein, redundant knowledge is merged by establishing "*identical_to*" relationships between duplicate entities.

For DR extracted from various documents, knowledge fusion means integrating these DRs by establishing interconnected relationships. We adopt two approaches to accomplish this task. One is to establish relations through the semantic similarity analysis of DR texts.

The semantic similarity of DR texts is used for the identification of duplicated entities. This approach is especially suitable for the fusion of DR sentences, such as *issues* and *intents*. The other method is to establish relationships through the metadata of documents, including citations, co-authors, co-affiliations, and others.

The semantic similarity between each pair of text ($T1$, $T2$), including words, phrases, clauses, or sentences, could be analyzed through the cosine similarity calculations between them [62,69]. The sentence semantic similarity analysis can be used to build relations between different documents, or within a document. The text pair ($T1$, $T2$) is represented by the corresponding vector embeddings (V_{t1}, V_{t2}), and the text similarity is calculated via cosine similarity based on vector embedding features using Formula (25):

$$\mathrm{Sim}(T_1,\ T_2) = \cos(\theta) = \frac{V_{t1} \times V_{t2}}{||V_{t1}||\ ||V_{t2}||} = \frac{\sum_{i=1}^{n} V_{t1i} * V_{t2i}}{\sqrt{\sum_{i=1}^{n} {V_{t1i}}^2}\ \sqrt{\sum_{i=1}^{n} {V_{t2i}}^2}} \tag{25}$$

where V_{t1i} and V_{t2i} are the i^{th} constituent elements of the V_{t1} and V_{t2} vectors, respectively.

4. Empirical Study

We set up experiments on the construction of knowledge networks, where knowledge is extracted from patent documents and academic paper documents, verifying the feasibility and performance of the proposed method. The experiment was conducted within the prototype we developed using CentOS Stream, Python, wxGlade-1.0.2, Scrapy, Neo4j Community 4.2, Java, Stanford Stanza, NLTK, Transformers, Neovis.js, and others. The flowchart of the experiments is shown in Figure 7. The experiment consists of three phases, namely, data preparation, entity and relationship extraction, and knowledge fusion.

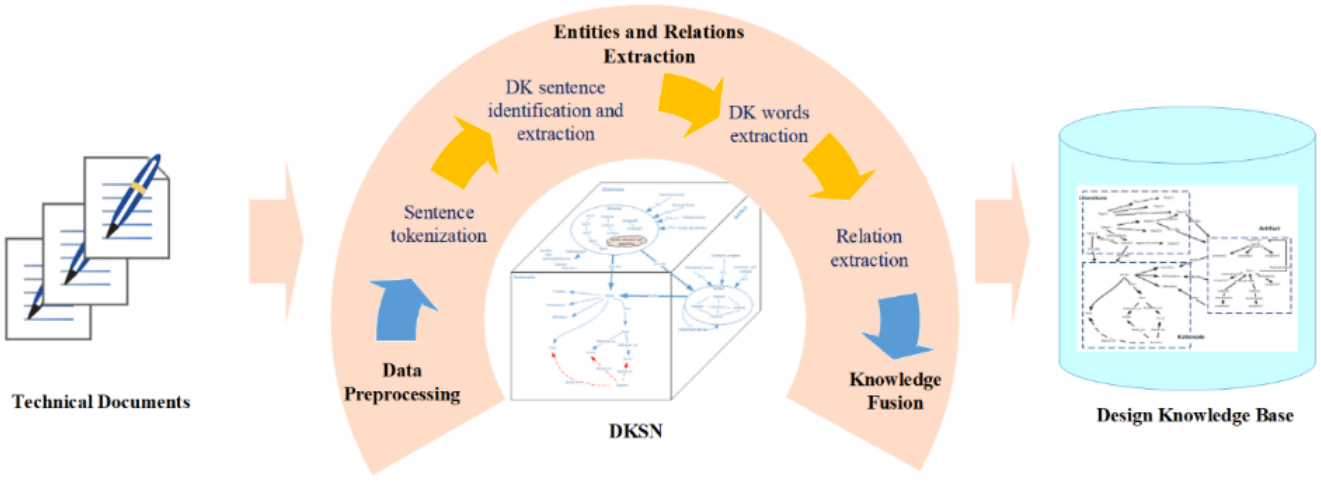


Figure 7. The flowchart of the DKN construction.

4.1. Data Preparation

The literature used for experiments includes 1491 patent documents on unmanned aerial vehicles, 1798 patent documents on rolling bearings, and 359 academic articles on unmanned aerial vehicles. Specifically, 1491 patent documents were collected from the United States Patent Office (USPTO) using "unmanned aerial vehicles" and "safety" as search keywords. Considering the large length of the patent text, only some of the contents were selected as sample data, including patent ID, title, abstract, description, background, and summary. Using "rolling bearing" as the search keyword, 1798 patent documents were collected from the USPTO. The extracted patent metadata included patent ID, title, inventor, inventor list, applicant, city of applicant, state of applicant, country of applicant, date filed, patentee, citation ID list, citation name list, and date published; and the content data included abstract, description, background, and summary. In total, 359 unmanned aerial

vehicle-related academic articles were selected from open access articles as research data. The extracted metadata from articles included the identifier and title, and the extracted content data included the abstract, introduction, discussion, conclusion, and body content. Furthermore, to establish a reference product data dictionary, we imported the data from ISO/TS 23768-1 on the definitions of rolling bearing classes and the definitions of rolling bearing properties.

4.2. Entities and Relations Extraction

Sentence-level DR is the basic unit extracted from a document. With the NLP tools, technical documents are tokenized as a list of sentences. The DR sentences are identified and extracted from the sentence list, including *issue* sentences, *intent* sentences, *artifact* sentences, *argument* sentences, *alternative* sentences, and others.

The experiment showed that when the training data exceeded a certain number (500) of sentences, the accuracy and recall of DR sentence identification based on BERT were significantly improved. To this end, we used the LAMB optimizer [70] for faster training, and sentence classification based on BERT [71] to identify and to extract DR sentences. A method based on lexical-syntactic patterns was used to realize the extraction of *artifact entities* and *artifact elements*, relations, *pros* and *cons*, and other feature words. Figure 8 shows an example of *artifact* extraction based on lexical-syntactic patterns and dependency parsers. Figure 9 shows the DKN extracted from a patent.

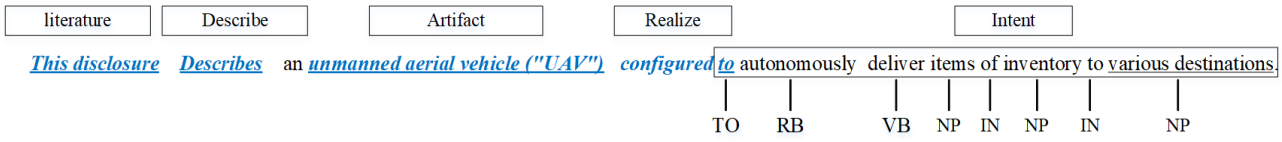


Figure 8. Example of artifact aggregation relations extraction.

4.3. Knowledge Fusion

As mentioned above, knowledge fusion within a document involves the merging of duplicate DR entities, and knowledge fusion between documents involves the construction of an interconnected knowledge network by establishing relationships. In the experiment, the object of knowledge fusion was aimed at not only different DR types, such as *issues* fusion and *artifacts* fusion, but also different text forms, such as sentence-level DR fusion and word-level DR fusion. For example, an entire sentence containing an *intent* is considered to be an *intent* sentence, and an *intent* phrase in this sentence is also seen as a more precise *intent*. Both patterns may exist simultaneously in the DKN. Similarly, a sentence containing *artifacts* is an *artifact* sentence, from which the *artifact entities* and *artifact elements* extracted can more accurately represent *artifacts*. In the experiment, the fusion of *intent* was focused on sentence-level fusion. The *artifact* fusion was focused on word-level entity alignment, which was used to build an *artifact* hierarchy system.

4.3.1. Knowledge Fusion within a Document

Intent sentence merging. The semantics of an *intent* sentence depends on its context. If a design *intent* sentence contains the same artifact words and the same *intent* phrases of each constituent *artifact*, then the two design *intent* sentences are considered to be identical. As shown in Figure 10, an "identical_to" relationship can be established between the two *intent* sentence nodes via a simple comparison of the duplicated *artifact* words and *intent* phrases. Furthermore, the cosine similarity analysis of two *intent* sentences is an effective approach for an *intent* sentence merge.

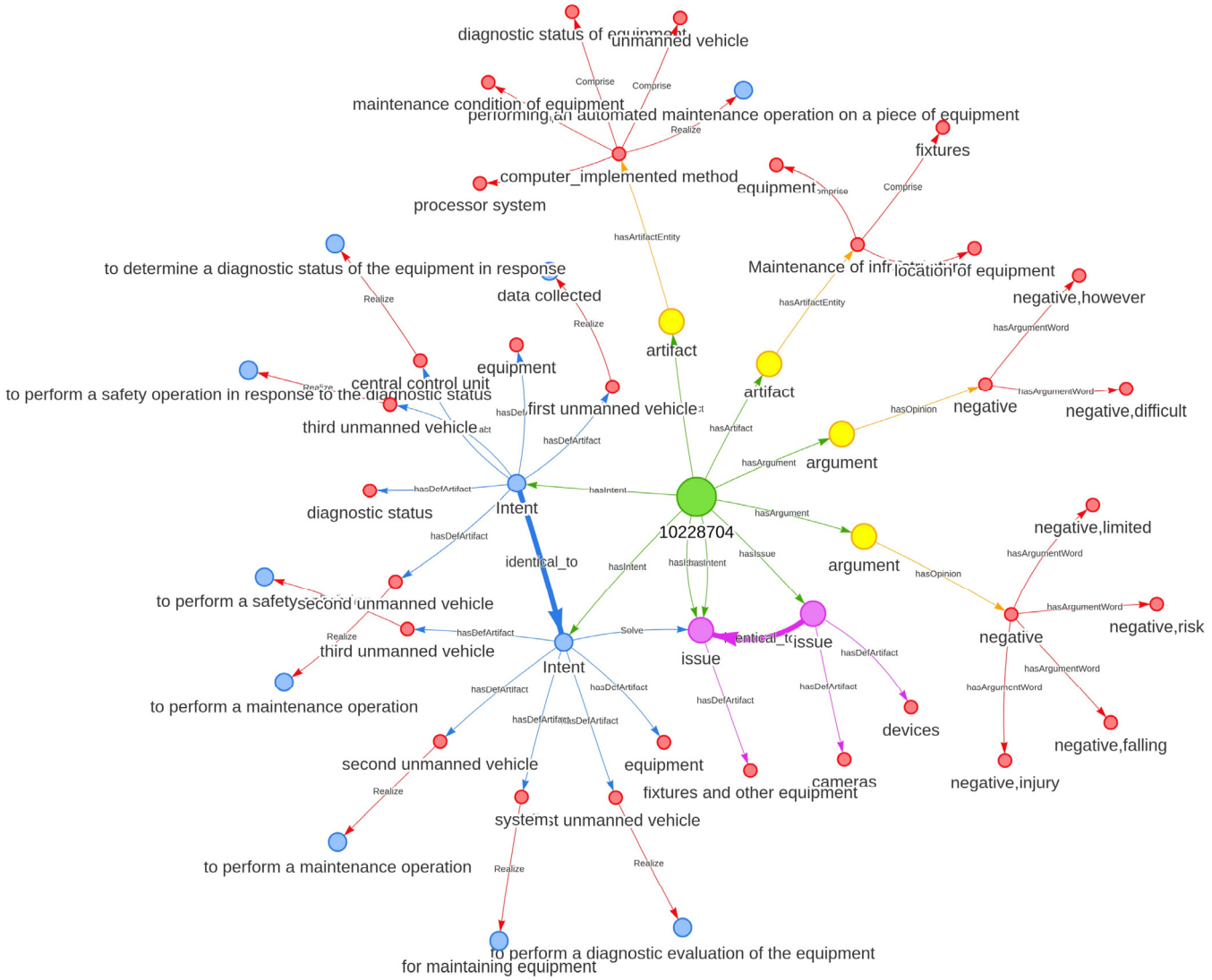


Figure 9. An example of a DKN extracted from the patent document. In the figure, the big node in the center (10228704) represents the patent, and 10228704 is the patent number; middle sized nodes (*artifact*, *argument*, *issue*, *intent*) are DR sentence nodes. Other small nodes are feature word nodes, such as *artifact entities*, *artifact elements*, *intent*, *pros* and *cons*, and others.

Issue sentence merging. For the patent documents, we assume that the same patent can only solve one design problem. If two *issues* are described in the same BDU, the two *issues* can be considered as the same one and can be combined, as shown in the first two rows of Table 5. Furthermore, a cosine similarity analysis-based knowledge merge of two *issue* sentences is also an effective approach.

Artifact word merging. The *artifact entities* and *artifact elements* are represented as words, and the *artifact* words in different *literature* express various semantics, depending on the context. This makes it difficult to determine whether two feature words are identical, through semantic similarity analysis alone. Therefore, *artifact* merging only applies to *artifacts* in the same document, and *artifacts* between different documents should not be merged. *Artifact* merging is performed by comparing whether the feature words of the two *artifacts* represent the same one. As shown in Figure 11, a system is described in the patent name "Methods and systems of an all-purpose broadband network", including a "local optimization server", and a "wireless RF access". The "local optimization server" includes "uplinks", "mobile devices", "part on air interface data rate priority value", "scheduler", and others. At the same time, the "wireless radio frequency access" includes a "scheduler

that provides an uplink" and a "downlink air interface access to mobile devices". The *artifact entity* "local optimization server" in the sentence node (10311) is identical to the *artifact element* "local optimization server" in the sentence node (10362).

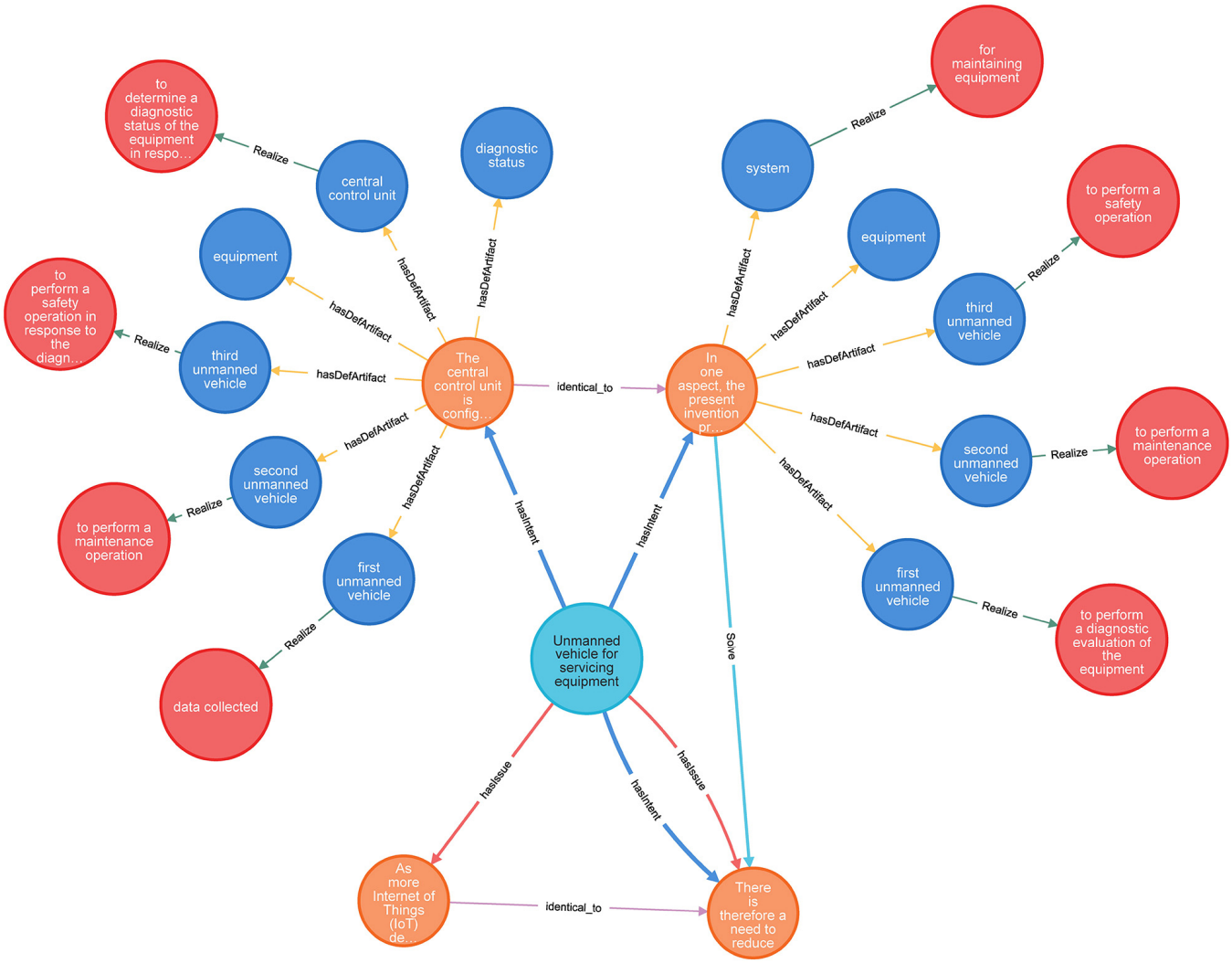


Figure 10. Example of fusion of design intent based on literature structure.

Table 5. Semantic similarity analysis for *issue* sentence fusion.

Relation	Source	Issue
issueIdenticalTo	Issue 1 in "summary" section	A system and method disclosed herein for providing communications security for small, low-powered systems that are actively and securely controlled over a wireless data link addresses the problems faced by prior art systems.
	Issue 2 in "background" section	There is therefore a need for a small, low-power, security-certifiable system for controlling actively guided ordinance.

Table 5. *Cont.*

Relation	Source	Issue
issueSimilarTo	9194431	It is this problem that the invention more particularly intends to address, by proposing a new suspension thrust bearing that is simple and economical to manufacture, and to assemble at the same time as guaranteeing a high resistance of the shock absorber bump stop to transmitted shocks.
	9664233	The invention more particularly aims to resolve these problems by proposing a new engagement–disengagement, suspension, or steering release bearing having a greater resistance to a separating force of the rolling bearing rings, while remaining compact.
	9068593	Against this background, the object of the present invention is to provide a rolling bearing that meets the aforementioned requirements of effective cooling and damping, and that can be produced in a constructionally simple and cost-effective manner.
issueIdenticalTo	9574630	The applicant of the present invention has already proposed in JP 2009-197863A an electric linear motion actuator which is free of this problem, and which can sufficiently increase power without the need for a separate speed reduction mechanism, and thus which can be used in an electric disk brake system, of which the linear motion stroke is relatively small.
	9188183	The applicant of the present invention has already proposed in JP 2009-197863A an electric linear motion actuator which is free of this problem, and which can sufficiently increase power without the need for a separate speed reduction mechanism and thus can be used in an electric disk brake system, of which the linear motion stroke is relatively small.
	9435411	In order to avoid this problem, the applicant of this invention proposed, in JP Patent Publication 2010-65777A and JP Patent Publication 2010-90959A, electric linear motion actuators which are suitable for use in electric disk brake systems, because they can increase power to a considerable degree without the need for a speed reduction mechanism, and are small in linear motion stroke.
	9182021	The applicant of the present invention has already proposed in JP 2010-65777A and JP 2010-90959A electric linear motion actuators which are free of this problem, and which can sufficiently increase power without the need for a separate reduction mechanism, and thus which can be used in an electric disk brake system, of which the linear motion stroke is relatively small.

However, if the same design document contains multiple sets of *artifact entities* and *artifact elements*, although the extracted names are the same, the meanings are different. This requires the comparison of two sets of *artifact elements* to distinguish whether the *artifact entities* are the same. As shown in Figure 12, both node 445 and node 446 have a "set of flight_status detecting sensors" *ArtifactEntity*. However, they are different because of their different *artifactElements*.

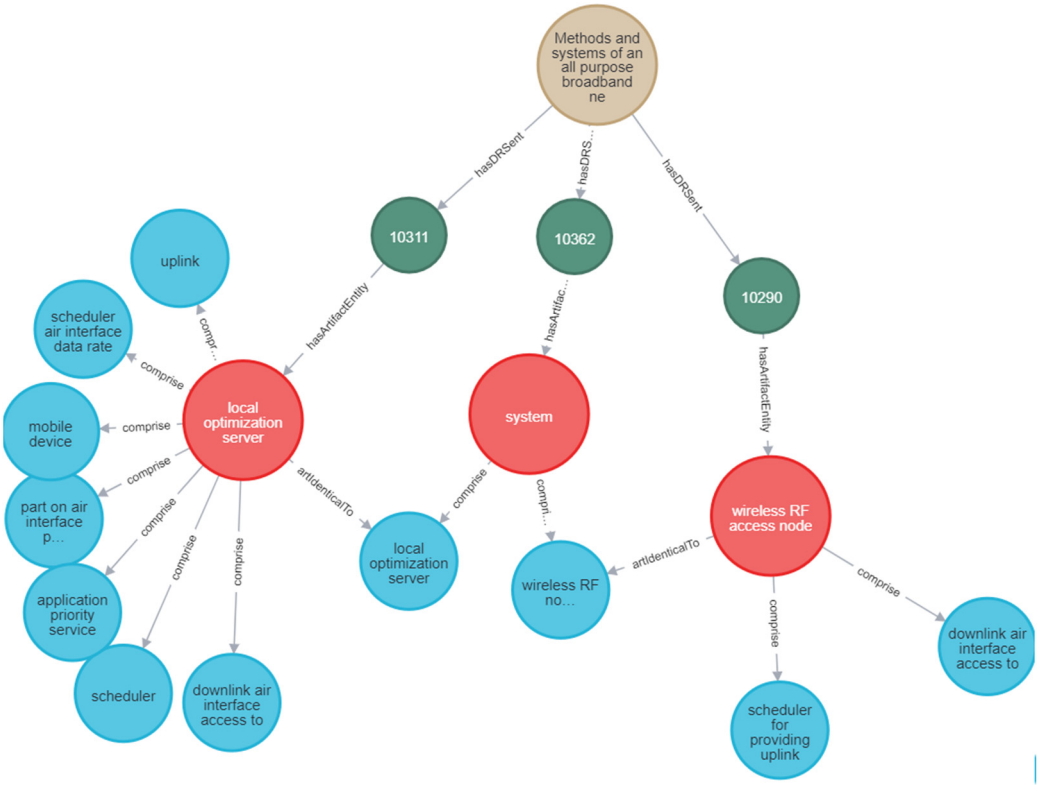


Figure 11. Artifact fusion between artifact entities and elements.

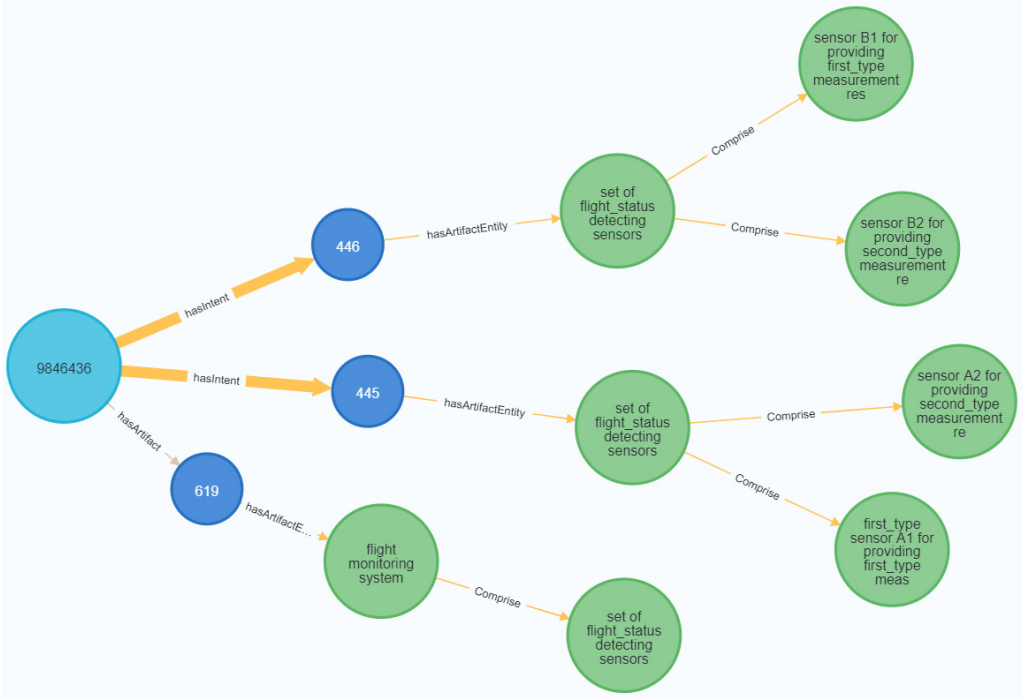


Figure 12. The same artifact entity with different meanings.

4.3.2. Knowledge Fusion between Documents

Knowledge fusion based on semantic similarity analysis. We use the deep learning model "sentence-transformers", specifically the "all_MiniLM-L12-V2" model, to analyze the semantic similarity of sentence pairs. A total of 747 *issue* sentences were analyzed by pairs, and a total of C_{747}^{2} = 278,631 similarity calculations were carried out. Through

the comparison of iterative loops, a total of 1394 pairs of "*issueSimilarTo*" relations and 237 groups of "*issueIdenticalTo*" relations were identified.

If the similarity coefficient (I_{sim}) is greater than 0.8, it is considered to be identical, and the "*issueIdenticalTo*" relationship is established. If the I_{sim} is greater than 0.6 and less than 0.8, it is a similar relationship, and the "*issueSimilarTo*" relation is established, as shown in Table 5.

4.4. Comparison and Discussion

To demonstrate the contributions of the proposed approach, we compared it with several typical state-of-the-art DR representations, as shown in Table 6. By comparison, the proposed approach has the following advantages: (1) It supports the classic DR theories. The DKSN representation is fully compatible with DR, based on design argumentation and design intent based on design thinking, and is more adaptable. DR is valuable because it is tacit knowledge about the process of design argumentation or design thinking. Most of the DR is lost, and only a small part of the knowledge is recorded in the documentation. Therefore, the extracted DR should be consistent with classical DR theory, and the most important information, such as *pros and cons*, and *alternatives*, should not be omitted. (2) It supports the NLP. The experimental results show that our method takes into consideration the linguistic features of the design literature, and is more suitable for sentence classification, sentiment analysis, information extraction, and other NLP technologies. For other studies on DR representation, only the ISAL model covers text mining. (3) It supports the construction of knowledge graphs. The construction of the design knowledge graphs would be beneficial to knowledge acquisition, sharing, and reuse. Based on the Semantic Web, ontology, and knowledge graph, a formalized ODKN could be established, which is suitable for the semantic retrieval and straightforward understanding of designers. The ODKN could accelerate the acquisition, sharing, and reuse of design knowledge. (4) It can provide designers with creative design stimuli. For innovative designers, DR should be presented in a short, easy-to-understand form. If the extracted DR is a long paragraph, it is not simple enough, and will hinder the designer's quick understanding. The DR extracted by our method is a DR sentence, phrase, or word, which meets this requirement.

Table 6. Comparisons of DR representation in the field of engineering design.

Literature Source	Info of Argumentation or Thinking	Support Text Mining	Straightforward and Intuitive	Support Knowledge Graph
[35]	issue, solution, artifact, argument	No	yes (term)	No
[20]	issue, solution (usage, performance, accuracy, product, material, structure, function)	N/A	Yes (term)	Yes
[42]	part, assembly, functional requirement, geometry feature, constraint	No	N/A	No
[43]	artifact, design history, participants, artifact structure, boundary representation	No	Yes (sentence)	yes
[34,72]	ISAL: issue, solution, artifact	Yes	No (paragraph)	No
Our approach	artifact, issue, intent, pros and cons, alternative, and relation	Yes	Yes (term, phrase, clause, and sentence)	Yes

To illustrate the metric of our DKSN model, we intend to conceptually compare DKSN with ISAL. ISAL is the only representation model of DR which is intended for DR discovery from patent documents. Figure 9 shows an example of extracted knowledge from a patent document using our DKSN model through text mining. Table 7 illustrates

the DR of the same patent, captured manually via reference to the methods in [34,72]. From the comparison, we can know that the extracted knowledge using DKSN could more accurately show the internal logical relationship of the extracted knowledge, and that it is more straightforward and intuitive, which is suitable for a designer as a kind of inspirational stimulus.

Table 7. A comparative example of DR representation using ISAL.

Issue:	
Maintenance of infrastructure that includes fixtures or other equipment can be difficult, depending on the location of the equipment. . . . A service person's limited maneuverability when on a ladder increases the risk of falling or serious injury during a maintenance operation. However, fixtures and other equipment in areas that are difficult to access are prevalent, not only in residential houses, but also in public buildings and public spaces. In other instances, the equipment needs to be powered down before a human can safely perform maintenance thereon. . . . There is also the issue of scale. As more Internet of Things (IoT) devices such as cameras, chemical sensors, etc., are deployed, there is a need to automate the maintenance of such IoT devices.	
Solution:	
The present invention provides a system for maintaining equipment within a predetermined area, including a first unmanned vehicle configured to perform a diagnostic evaluation of the equipment, a second unmanned vehicle configured to perform a maintenance operation, and a third unmanned vehicle configured to perform a safety operation. . . . A central control unit is operably coupled to the first unmanned vehicle, the second unmanned vehicle, and the third unmanned vehicle. The central control unit is configured to determine a diagnostic status of the equipment in response to data collected by the first unmanned vehicle, and to dispatch at least one of the second unmanned vehicles, to perform a maintenance operation, and the third unmanned vehicle, to perform a safety operation in response to the diagnostic status of the equipment.	
Artifact:	
first unmanned vehicle	second unmanned vehicle
third unmanned vehicle	central control unit

5. Conclusions

To promote the acquisition, sharing, and reuse of design knowledge in technical literature, and to provide inspirational stimuli for designers, we propose a DR-centric DKSN metamodel and related knowledge fusion approach, which can integrate the design object knowledge network and the literature network, and which can build a three-dimensional ODKN. This could solve the problem of DR fragmentation.

The DKSN metamodel covers the DKN within a document and the ODKN between documents, which includes knowledge networks in three dimensions. The knowledge network within a technical document is established based on document writing rules and syntactic structure. ODKN is constructed with publicly available technical documents through citations and other metadata relations. The artifact network is established by mapping the extracted *artifacts* with the reference product data dictionary. The reference product data dictionary defines the hierarchy of product classes and properties. In addition, knowledge fusion based on document structure, semantic similarity analysis, and citations are used to achieve knowledge fusion, reduce knowledge redundancy, and establish knowledge networks between documents.

Based on the DKSN method, we set up an experiment to automatically extract entities, relationships, and attributes from academic articles and patent documents. The rolling bearing product data dictionary from ISO/TS 23768.1 was imported as a reference artifact network. The knowledge fusion of entities such as *literature*, *artifacts*, *issues*, and *intents* is carried out, and the effectiveness and limitations of the knowledge fusion are verified.

One of the contributions of this paper is that a DKSN metamodel is used for the internalization of explicit design knowledge, taking specific sentences and words in the literature as the nodes of the DKN, which is conducive to the rapid understanding and

mastery of designers. At the same time, the proposed method is suitable for the processing of technical documents, knowledge extraction, and network construction.

The limitations of this study are that the research objects are mainly patent documents and journal articles. The research hypothesis is that there is usually only one clear design issue in this type of technical literature, around which the design intention is proposed, the design object is elaborated, and the pros and cons are discussed. This assumption may not be true for other technical literature. For example, technical literature such as design specifications and review reports within the company may contain multiple technical issues. In the future, to construct ODKNs for various technical literature, it is necessary to carry out deeper knowledge fusion research.

Author Contributions: Conceptualization, J.L.; methodology, writing—original draft, G.Y.; writing—review and editing, Q.Z. and Y.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key Research and Development Program from the Ministry of Science and Technology of the Republic of China under Grant (2021YFB1716201, 2022YFF0605201).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank all anonymous reviewers and editors for their helpful suggestions for the improvement of this paper.

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

References

1. *BS 7000-1:2008*; Design Management Systems—Part 1: Guide to Managing Innovation. British Standards Institution: London, UK, 2008.
2. Kodama, M. Boundaries Knowledge (Knowing)—A Source of Business Innovation. *Knowl. Process Manag.* **2019**, *26*, 210–228. [CrossRef]
3. Rob, B.; Ken, W.; Michael, M.; David, K. Capturing Design Rationale. *Comput.-Aided Des.* **2009**, *41*, 173–186.
4. Setchi, R.; Bouchard, C. In Search of Design Inspiration: A Semantic-Based Approach. *J. Comput. Inf. Sci. Eng.* **2010**, *10*. [CrossRef]
5. Liu, L.; Li, Y.; Xiong, Y.; Cavallucci, D. A New Function-Based Patent Knowledge Retrieval Tool for Conceptual Design of Innovative Products. *Comput. Ind.* **2020**, *115*, 103154. [CrossRef]
6. Valverde, U.Y.; Nadeau, J.-P.; Scaravetti, D. A New Method for Extracting Knowledge from Patents to Inspire Designers during the Problem-Solving Phase. *J. Eng. Des.* **2017**, *28*, 369–407. [CrossRef]
7. Goucher-Lambert, K.; Gyory, J.T.; Kotovsky, K.; Cagan, J. Adaptive Inspirational Design Stimuli: Using Design Output to Computationally Search for Stimuli That Impact Concept Generation. *J. Mech. Des.* **2020**, *142*. [CrossRef]
8. Song, B.; Srinivasan, V.; Luo, J. Patent Stimuli Search and Its Influence on Ideation Outcomes. *Des. Sci.* **2017**, *3*, e25. [CrossRef]
9. Eckert, C.; Stacey, M. Sources of Inspiration: A Language of Design. *Des. Stud.* **2000**, *21*, 523–538. [CrossRef]
10. Dybvik, H.; Abelson, F.G.; Aalto, P.; Goucher-Lambert, K.; Steinert, M. Inspirational Stimuli Improve Idea Fluency during Ideation: A Replication and Extension Study with Eye-Tracking. *Proc. Des. Soc.* **2022**, *2*, 861–870. [CrossRef]
11. Hahm, G.J.; Lee, J.H.; Suh, H.W. Semantic Relation Based Personalized Ranking Approach for Engineering Document Retrieval. *Adv. Eng. Inform.* **2015**, *29*, 366–379. [CrossRef]
12. Peng, G.; Wang, H.; Zhang, H.; Huang, K. A Hypernetwork-Based Approach to Collaborative Retrieval and Reasoning of Engineering Design Knowledge. *Adv. Eng. Inform.* **2019**, *42*, 100956. [CrossRef]
13. Thomas, R.G.; Daniel, R. *Design Knowledge and Design Rationale: A Framework for Representation, Capture, and Use*; Knowledge Systems Laboratory, Computer Science Department, Stanford University: Stanford, CA, USA, 1991.
14. Dworschak, F.; Kügler, P.; Schleich, B.; Wartzack, S. Model and Knowledge Representation for the Reuse of Design Process Knowledge Supporting Design Automation in Mass Customization. *Appl. Sci.* **2021**, *11*, 9825. [CrossRef]
15. Yue, G.; Liu, J.; Hou, Y. *Design Rationale Knowledge Management: A Survey*; Luo, Y., Ed.; Springer International Publishing: Cham, Switzerland, 2018; pp. 245–253.
16. Fischer, G.; Shipman, F. Collaborative Design Rationale and Social Creativity in Cultures of Participation. In *Creativity and Rationale: Enhancing Human Experience by Design*; Carroll, J.M., Ed.; Springer: London, UK, 2013; pp. 423–447, ISBN 978-1-4471-4111-2.
17. Morteza, P.; Joel, J.; Fredrik, E. Capturing, Structuring and Accessing Design Rationale in Integrated Product Design and Manufacturing Processes. *Adv. Eng. Inform.* **2016**, *30*, 522–536.

18. Mandorli, F.; Borgo, S.; Wiejak, P. From Form Features to Semantic Features in Existing MCAD: An Ontological Approach. *Adv. Eng. Inform.* **2020**, *44*, 101088. [CrossRef]
19. Gopsill, J.A.; McAlpine, H.C.; Hicks, B.J. A Social Media Framework to Support Engineering Design Communication. *Adv. Eng. Inform.* **2013**, *27*, 580–597. [CrossRef]
20. Jia, J.; Zhang, Y.; Saad, M. An Approach to Capturing and Reusing Tacit Design Knowledge Using Relational Learning for Knowledge Graphs. *Adv. Eng. Inform.* **2022**, *51*, 101505. [CrossRef]
21. Raymond, C.W.S.; James, M.R.; Theodore, L.; Zoe, K. Automated Generation of Engineering Rationale, Knowledge and Intent Representations during the Product Life Cycle. *Virtual Real.* **2012**, *16*, 69–85.
22. Hiroyuki, Y. Systematization of Design Knowledge. *CIRP Ann.* **1993**, *42*, 131–134.
23. Bijl-Brouwer, M.V.D.; Malcolm, B. Systemic Design Principles in Social Innovation: A Study of Expert Practices and Design Rationales. *She Ji J. Des. Econ. Innov.* **2020**, *6*, 386–407. [CrossRef]
24. Conklin, E.J.; Yakemovic, K.B. A Process-Oriented Approach to Design Rationale. *Hum.-Comput. Interact.* **1991**, *6*, 357–391.
25. Frank, M.S.I.; Catherine, C.M. Formality Considered Harmful: Experiences, Emerging Themes, and Directions on the Use of Formal Representations in Interactive Systems. *Comput. Support. Coop. Work* **1999**, *8*, 333–352.
26. Lee, J. *Decision Representation Language (DRL) and Its Support Environment*; MIT Artificial Intelligence Laboratory Working Papers, WP-325; MIT Artificial Intelligence Laboratory: Cambridge, MA, USA, 1989.
27. Jeff, C.; Michael, L.B. GIBIS: A Hypertext Tool for Exploratory Policy Discussion. *ACM Trans. Inf. Syst.* **1988**, *6*, 303–331.
28. Wang, H.; Johnson, A.L.; Bracewell, R.H. The Retrieval of Structured Design Rationale for the Re-Use of Design Knowledge with an Integrated Representation. *Adv. Eng. Inform. Based Eng. Support Complex Prod. Des.* **2012**, *26*, 251–266. [CrossRef]
29. Hao, J.; Zhao, L.; Milisavljevic-Syed, J.; Ming, Z. Integrating and Navigating Engineering Design Decision-Related Knowledge Using Decision Knowledge Graph. *Adv. Eng. Inform.* **2021**, *50*, 101366. [CrossRef]
30. Ferraro, G.; Wanner, L. Towards the Derivation of Verbal Content Relations from Patent Claims Using Deep Syntactic Structures. *Knowl. -Based Syst.* **2011**, *24*, 1233–1244. [CrossRef]
31. Rogers, B.; Qiao, Y.; Gung, J.; Mathur, T.; Burge, J.E. *Using Text Mining Techniques to Extract Rationale from Existing Documentation*; Gero, J.S., Hanna, S., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 457–474.
32. Lester, M.; Guerrero, M.; Burge, J. Using Evolutionary Algorithms to Select Text Features for Mining Design Rationale. *Artif. Intell. Eng. Des. Anal. Manuf.* **2020**, *34*, 132–146. [CrossRef]
33. Kurtanović, Z.; Maalej, W. On User Rationale in Software Engineering. *Requir. Eng.* **2018**, *23*, 357–379. [CrossRef]
34. Liu, Y.; Liang, Y.; Kwong, C.K.; Lee, W.B. A New Design Rationale Representation Model for Rationale Mining. *J. Comput. Inf. Sci. Eng.* **2010**, *10*. [CrossRef]
35. Zhang, Y.; Luo, X.; Li, J.; Buis, J.J. A Semantic Representation Model for Design Rationale of Products. *Adv. Eng. Inform.* **2013**, *27*, 13–26. [CrossRef]
36. Horst, W.J.R.; Melvin, M.W. Dilemmas in a General Theory of Planning. *Policy Sci.* **1973**, *4*, 155–169. [CrossRef]
37. Kunz, W.; Rittel, H.W.J. *Issues as Elements of Information Systems*; Institute of Urban and Regional Development, University of California: Oakland, CA, USA, 1970.
38. Allan, M.; Richard, Y.; Victoria, B.; Thomas, M. Questions, Options, and Criteria: Elements of Design Space Analysis. *Hum.-Comput. Interact.* **1991**, *6*, 201–250.
39. de Medeiros, A.P.; Schwabe, D. Kuaba Approach: Integrating Formal Semantics and Design Rationale Representation to Support Design Reuse. *Artif. Intell. Eng. Des. Anal. Manuf.* **2008**, *22*, 399–419. [CrossRef]
40. Tetsuo, T. From General Design Theory to Knowledge-Intensive Engineering. *Artif. Intell. Eng. Des. Anal. Manuf.* **1994**, *8*, 319–333.
41. Liu, J.; Hu, X. A Reuse Oriented Representation Model for Capturing and Formalizing the Evolving Design Rationale. *Artif. Intell. Eng. Des. Anal. Manuf.* **2013**, *27*, 401–413. [CrossRef]
42. Cheng, Z.; Ma, Y. A Functional Feature Modeling Method. *Adv. Eng. Inform.* **2017**, *33*, 1–15. [CrossRef]
43. Cheng, Y.; He, F.; Lv, X.; Cai, W. On the Role of Generating Textual Description for Design Intent Communication in Feature-Based 3D Collaborative Design. *Adv. Eng. Inform.* **2019**, *39*, 331–346. [CrossRef]
44. Lee, J.; Lai, K.-Y. What's in Design Rationale? *Hum.-Comput. Interact.* **1991**, *6*, 251–280.
45. Kenett, Y.N.; Faust, M. A Semantic Network Cartography of the Creative Mind. *Trends Cogn. Sci.* **2019**, *23*, 271–274. [CrossRef]
46. Kenett, Y.N.; Anaki, D.; Miriam, F. Investigating the Structure of Semantic Networks in Low and High Creative Persons. *Front. Hum. Neurosci.* **2014**, *8*, 407. [CrossRef]
47. Liu, Q.; Wang, K.; Li, Y.; Chen, C.; Li, W. A Novel Function-Structure Concept Network Construction and Analysis Method for a Smart Product Design System. *Adv. Eng. Inform.* **2022**, *51*, 101502. [CrossRef]
48. Siemens, G. Knowing Knowledge; Lulu.com. 2006. ISBN 978-1-4303-0230-8. Available online: https://www.knowingknowledge.com/ (accessed on 12 January 2023).
49. Johansson, J.; Contero, M.; Company, P.; Elgh, F. Supporting Connectivism in Knowledge Based Engineering with Graph Theory, Filtering Techniques and Model Quality Assurance. *Adv. Eng. Inform.* **2018**, *38*, 252–263. [CrossRef]
50. Sarica, S.; Luo, J. Design knowledge representation with technology semantic network. *Proc. Des. Soc.* **2021**, *1*, 1043–1052. [CrossRef]
51. Loureiro, G.B.; Ferreira, J.C.E.; Messerschmidt, P.H.Z. Design Structure Network (DSN): A Method to Make Explicit the Product Design Specification Process for Mass Customization. *Res. Eng. Des.* **2020**, *31*, 197–220. [CrossRef]

52. Chen, H.; Luo, X. An Automatic Literature Knowledge Graph and Reasoning Network Modeling Framework Based on Ontology and Natural Language Processing. *Adv. Eng. Inform.* **2019**, *42*, 100959. [CrossRef]
53. Zhao, D.; Andreas, S. *Analysis and Visualization of Citation Networks*, Synthesis Lectures on Information Concepts, Retrieval, and Services, 1st ed.; Springer: Cham, Switzerland, 2015; ISBN 978-3-031-02291-3.
54. Li, X.; Jiang, Z.; Song, B.; Liu, L. Long-Term Knowledge Evolution Modeling for Empirical Engineering Knowledge. *Adv. Eng. Inform.* **2017**, *34*, 17–35. [CrossRef]
55. Rathore, M.M.U.; Gul, M.J.J.; Paul, A.; Khan, A.A.; Ahmad, R.W.; Rodrigues, J.J.P.C.; Bakiras, S. Multilevel Graph-Based Decision Making in Big Scholarly Data: An Approach to Identify Expert Reviewer, Finding Quality Impact Factor, Ranking Journals and Researchers. *IEEE Trans. Emerg. Top. Comput.* **2021**, *9*, 280–292. [CrossRef]
56. Singhal, A. 2012 Introducing the Knowledge Graph: Things, Not Strings. Available online: https://www.blog.google/products/search/introducing-knowledge-graph-things-not/ (accessed on 4 November 2021).
57. Hogan, A.; Blomqvist, E.; Cochez, M.; Amato, C.D.; Melo, G.D.; Gutierrez, C.; Kirrane, S.; Gayo, J.E.L.; Navigli, R.; Neumaier, S.; et al. Knowledge Graphs. *ACM Comput. Surv.* **2021**, *54*, 1–37. [CrossRef]
58. Nguyen, H.L.; Vu, D.T.; Jung, J.J. Knowledge Graph Fusion for Smart Systems: A Survey. *Inf. Fusion* **2020**, *61*, 56–70. [CrossRef]
59. Liu, M.; Li, X.; Li, J.; Liu, Y.; Zhou, B.; Bao, J. A Knowledge Graph-Based Data Representation Approach for IIoT-Enabled Cognitive Manufacturing. *Adv. Eng. Inform.* **2022**, *51*, 101515. [CrossRef]
60. Zhou, B.; Hua, B.; Gu, X.; Lu, Y.; Peng, T.; Zheng, Y.; Shen, X.; Bao, J. An End-to-End Tabular Information-Oriented Causality Event Evolutionary Knowledge Graph for Manufacturing Documents. *Adv. Eng. Inform.* **2021**, *50*, 101441. [CrossRef]
61. Ringsquandl, M.; Lamparter, S.; Lepratti, R.; Kröger, P.; Lödding, H. *Knowledge Fusion of Manufacturing Operations Data Using Representation Learning*; Lödding, H., Riedel, R., Thoben, K.-D., von Cieminski, G., Kiritsis, D., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 302–310.
62. Ji, Y.; Hu, L.; Liu, S.; Xu, Z.; Liu, Y.; Liu, K.; Tang, S.; Liu, Q.; Xiao, W. MEFE: A Multi-FEature Knowledge Fusion and Evaluation Method Based on BERT. In *Proceedings of the Algorithms and Architectures for Parallel Processing*; Qiu, M., Ed.; Springer International Publishing: Cham, Switzerland, 2020; pp. 449–462.
63. Nonaka, I.; Takeuchi, H. *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*, 1st ed.; Oxford University Press: Oxford, UK, 1995.
64. Nonaka, I.; Toyama, R.; Hirata, T.; Bigelow, S.J.; Hirose, A.; Kohlbacher, F. The Theoretical Framework. In *Managing Flow: A Process Theory of the Knowledge-Based Firm*; Nonaka, I., Toyama, R., Hirata, T., Bigelow, S.J., Hirose, A., Kohlbacher, F., Eds.; Palgrave Macmillan UK: London, UK, 2008; pp. 18–52, ISBN 978-0-230-58370-2.
65. Dubberly, H.; Evenson, S. Design as Learning—Or "Knowledge Creation"—The SECI Model. *Interactions* **2011**, *18*, 75–79. [CrossRef]
66. Vătămănescu, E.-M.; Bratianu, C.; Dabija, D.-C.; Popa, S. Capitalizing Online Knowledge Networks: From Individual Knowledge Acquisition towards Organizational Achievements. *J. Knowl. Manag.* 2022, *ahead-of-print*. [CrossRef]
67. Simon, H.A. *The Sciences of the Artificial*; MIT Press: Cambridge, MA, USA, 1996; ISBN 0-262-19374-4.
68. Lee, J. Design Rationale Systems: Understanding the Issues. *IEEE Expert* **1997**, *12*, 78–85. [CrossRef]
69. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings Using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*; Association for Computational Linguistics: Hong Kong, China, November 2019; pp. 3982–3992.
70. You, Y.; Li, J.; Reddi, S.J.; Hseu, J.; Kumar, S.; Bhojanapalli, S.; Song, X.; Demmel, J.; Keutzer, K.; Hsieh, C.-J. Large Batch Optimization for Deep Learning: Training BERT in 76 Minutes. *arXiv* **2020**, arXiv:1904.00962.
71. 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, Volume 1 (Long and Short Papers)*; Association for Computational Linguistics: Minneapolis, MN, USA, June 2019; pp. 4171–4186.
72. Liang, Y.; Liu, Y.; Kwong, C.K.; Lee, W.B. Learning the "Whys": Discovering Design Rationale Using Text Mining—An Algorithm Perspective. *Comput. Aided Des.* **2012**, *44*, 916–930. [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.