From Dynamic to Evolvable Knowledge Graphs in Manufacturing: Systematic Literature Review on Learning Approaches

Anna Teern [a,*], Nada Elgendy [a], Markus Kelanti [a], Henna Tammia [a] and Tero Päivärinta [a,b]

[a] *M3S - Empirical Software Engineering on Software, Systems, and Services, University of Oulu, Oulu, Finland*
E-mail: anna.teern@oulu.fi E-mail: nada.sanad@oulu.fi E-mail: markus.kelanti@oulu.fi
E-mail: henna.tammia@oulu.fi E-mail: tero.paivarinta@oulu.fi
[b] *Information Systems, Luleå University of Technology, Luleå, Sweden*

Abstract. This systematic literature review investigates how evolvable KGs enhance manufacturing under Industry 4.0 and 5.0, identifying key technologies and gaps. Evolvable KGs adapt to changing knowledge by leveraging machine learning algorithms and human expertise to enhance decision-making, operational efficiency, and predictive maintenance capabilities beyond the capabilities of dynamic KGs. Despite the advancements, challenges persist in quality assurance, process planning, and the integration of human expertise. The findings advocate for addressing these issues to foster wider adoption and optimization of KG technologies in manufacturing. This review maps existing literature based on the KG construction process stages. The main results are the state-of-the-art tasks in creating evolvable KGs in manufacturing and categories of learning approaches in evolvable KGs. The results contribute by updating the KG construction process and widening the understanding of evolvable KGs that utilize learning. By deepening the understanding of how KGs can evolve, this review sets the base for future research to develop more dynamic and intelligent systems tailored to the emerging demands of Industry 5.0.

Keywords: intelligent system, knowledge graph, evolvable knowledge graph, knowledge engineering

1. Introduction

Types of knowledge graphs (KGs), initially conceptualized as networks of concepts in the 1960s [58], have long played a key role in representing human knowledge in a format that computers can process. A KG is a graph-based data model representing knowledge of the world; in this model, nodes represent important entities, and edges represent relationships between these entities [34]. The notation in the KG sets human knowledge into a computer-readable format, enabling the creation of knowledge-based systems and applications.

Now, KGs also enhance human understanding of data [2]. Knowledge presentation is fundamental for enabling intelligence augmentation (IA) applications [8], hybrid intelligence [14] or collaborative rationality [17], where machine and human capabilities are combined.

Data-intensive solutions for industries characterize Industry 4.0. Recently, Industry 5.0 has transitioned from this data-centric perspective to human involvement at different levels in the industry. With this aim of human-machine collaboration, KGs offer many practical solutions for companies [81].

*Corresponding author. E-mail: anna.teern@oulu.fi.

Today, KGs are prevalent across many domains and present in various applications. In manufacturing, KGs enable adhering to standards [19], enhance predictive maintenance through semantic digital twins [27], and may support sustainability through energy consumption optimization [47]. Despite the advances, research remains necessary to combine machine and human intelligence efficiently and to adapt to changing environments [9, 49].

Previous literature reviews have laid a foundation for understanding the KG construction process (KGCP) across various domains, such as [67, 69]. Section 1.2 describes the process further based on these two reviews. Furthermore, [9] reviews the application of KGs in manufacturing, and [68] defines the design objectives for evolvable KGs in manufacturing. Building upon these contributions, the review focuses on the evolvability of KGs in the manufacturing sector.

Evolving knowledge or KGs are usually described in terms of temporal variations of knowledge [46, 57], a significant aspect in dynamic environments such as manufacturing. However, the learning aspect is still overlooked in these descriptions of evolving knowledge. This study diverges from the typical focus on changing data to explore how KGs evolve through integrating machine learning techniques and human interaction.

Firstly, learning involves transforming experience into reusable knowledge for a new experience (adapted from [52]). When a KG represents knowledge, learning materializes in an evolvable KG. Therefore, an evolving KG is defined as a KG where new subjects, objects, or relations formed as [s, p, o] triples are added or removed. In a dynamic KG, only objects change due to the dynamic nature of the raw data.

The following sections describe the semantic web and linked data related to KGs in section 1.1 and the KGCP used in this paper in section 1.2.

1.1. Related concepts

After the advent of semantic networks, ontologies emerged, aligning more closely with the modern requirements of applications for knowledge representation. Ontologies organize domain knowledge by detailing relationships between concepts.

In 2006, the concept of linked data was introduced [33], permanently changing knowledge sharing and reuse. Based on interconnecting data across the web, this concept laid the foundation for the semantic web [31]. As organizations began to adopt linked data practices, they inadvertently contributed to forming the semantic web. One of the most notable contributions in this field was made by Tim Berners-Lee, who formulated the principles of linked data [7]. Additionally, the linked open data 5-star scheme, proposed in this context, became a benchmark for data sharing standards. This was later extended to a 7-star open data scheme [36].

The semantic web can be viewed as a field encompassing ontologies, linked data, and KGs or as a technology where data is structured in a graph format, enriched with semantic meaning, and made accessible on the web [33]. The semantic web and KG are identical in concepts and technology if viewed from the latter perspective. However, the semantic web was primarily driven by academia because it focused on creating an open data space on the web, and companies adopted the KG concept for closed knowledge sharing. For this reason, this study focuses on KG construction rather than semantic web or linked data.

1.2. Construction process stages and tasks

Building on previous research [67, 68], Fig. 1 illustrates the KG construction process. This process is described in five stages, each comprising a few tasks.

Creating a KG always begins from utilization perspectives and ends in utilizing stored knowledge. Hence, the process is visualized with a circular notation. The knowledge utilization is oriented towards the application-specific aspects of the stored knowledge. It considers factors such as application and updates, ensuring that the structure and content of the knowledge align with the intended use. In [67], identifying data begins the process, and updating knowledge under the "Maintain KG"-step ends it, recognizing a similar circular process.

The next phase, knowledge acquisition, involves three tasks: ontology selection, data acquisition, and knowledge extraction. [67] separates these into two steps. The ontology is selected if one exists or constructed if data is structured in one step. Otherwise, knowledge is extracted from unstructured data in another step. This study includes the previous tasks in one step because they all involve raw data; only the method differs based on the type of data.

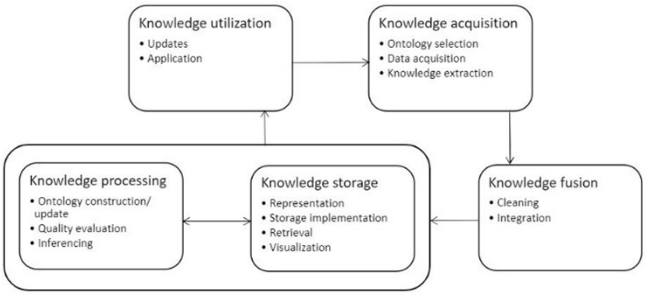

Fig. 1. KG construction process

Ontology selection is a human effort to identify and choose an appropriate ontology. Reference ontologies, for example, are reusable in a particular field or context [38]. Data acquisition involves gathering or linking data from heterogeneous sources. This is followed by knowledge extraction, where elements are typically extracted from data as triples (noun-verb-noun), representing the head-relationship-tail structure.

Following these, [67] describes the "Process Knowledge"-step, which includes integration and processing tasks. This study separated these tasks into different stages, as integration must be performed before storage activities, but knowledge can also be processed when stored. Therefore, the knowledge fusion stage follows knowledge acquisition, removes redundant elements, and resolves inconsistencies that often arise during integration. Diverging from data fusion, this process occurs on a more conceptual level.

Knowledge storage is the third stage, addressing all aspects of digitally storing knowledge. This encompasses considerations for knowledge representation, storage technologies, and methods for visualization and retrieval.

The fourth stage, knowledge processing, is central to enabling effective, further use of stored knowledge. It includes tasks such as conducting a quality check to remove erroneous knowledge, updating or constructing ontologies, and employing inferencing methods to discover new knowledge. [67] includes all these tasks in the process, but they are included in different steps: evaluation is conducted under "KG Maintenance", ontology is constructed either in the "Construct Ontology" or "Process Knowledge" step, and knowledge is completed (i.e., reasoned) in the "Process Knowledge" step. However, the order of the tasks is mostly the same in both processes.

This study aims to review the state-of-the-art (SotA) tasks contributing to evolvable KG in manufacturing. The study is a systematic literature review inspired by [16, 39]. It seeks to identify research gaps in the KG lifecycle and explore how learning is integrated into the construction and maintenance processes. This paper is structured to provide an overview of the development and application of evolvable KGs within the manufacturing sector. The following section describes the systematic approach employed to select and analyze relevant literature, detailing the search strategy, selection criteria, and analytical framework used. The paper's core, section 3, presents a thematic analysis of the findings organized around the KGCP stages. A discussion section 4 interprets these findings, drawing connections to existing literature, identifying research gaps, and suggesting avenues for future work. The paper concludes in section 5, summarizing the main contributions, implications for theory and practice, and the potential impact of evolvable KGs on advancing intelligent manufacturing systems.

2. Methodology

This section will outline the methodology implemented for the systematic literature review (SLR) following [39]. Articles on industrial applications in production and industrial automation focusing on KG implementation within manufacturing are sought. Managerial or enterprise-level solutions are not included.

2.1. Research Questions

Research questions investigated in the study have been selected for a design science research (DSR) effort to study knowledge engineering in manufacturing. The main research problem is how evolvable KGs are built in this field. The problem is studied using the following research questions (RQs):

Table 1
SEARCH STRATEGY

Search in fields	title, abstract, keywords
	full text in GS
	title in GS
Article types	journal papers
	conference papers
	workshop papers
	books
	book sections
Search engines	ACM Digital Library
	IEEExplore
	Web of Science
	Scopus, ScienceDirect
	Springer
	Google Scholar

- RQ 1: What is the TRL of the solutions?
- RQ 2: How are tasks automated?
- RQ 3: How does the KG change through the tasks?

Technical readiness level (TRL) has been used in many projects to guide development efforts from an idea to a complete system in operation [82]. RQ 1 gives an insight into how advanced the described solutions are in the literature. RQ 2 helps to identify what tasks can be automatized and how automatized the KGCP is. RQ 3 helps to understand how evolvability is considered in the KGCP. The results will provide insights into the current SotA of KGCP tasks and how the tasks support the evolvability of the KG.

As [40] suggests, the literature's population, intervention, comparison, outcomes, and context are now detailed. Based on the research questions mentioned above, the intervention of a study should be a KG or an industrial KG (IKG), and the outcome of an article should be a KGCP task completed in manufacturing. The context is (IKG), and the TRLs, task automation, and changes to the KG through tasks are compared.

2.2. Search Strategy

Based on the previous section, the databases for search and the search string are determined. Relevant search engines in this field include ACM Digital Library, IEEExplore, Web of Science, Scopus, Science Direct, and Google Scholar, based on a similar review [9]. Articles were extracted on March 15, 2022. The strategy is shown in table 1.

The search string is created based on the study's intervention, a KG, and the contextual term IKG, shown in table 2. Other search terms come from common concepts in the population, such as the manufacturing field in the era of Industry 4.0 and 5.0. The search is conducted on title, abstract, and keywords, except in Google Scholar (GS), as it is not possible due to a complex search term. Instead, only the first search term (first line on intervention) is used for full-text search, and the latter part is only searched in the title, which are the only options for targeting the search.

Table 2
SEARCH TERMS

INTERVENTION	"industrial knowledge graph" OR
	("knowledge graph" AND
POPULATION	(industry OR
	industrie OR
	"cyber physical system" OR
	"industrial internet of things" OR
	manufactur*))

Table 3

TECHNOLOGY READINESS LEVELS [82]

TRL 1	basic principles observed
TRL 2	technology concept formulated
TRL 3	experimental proof of concept
TRL 4	technology validated in the lab
TRL 5	technology validated in a relevant environment
TRL 6	technology demonstrated in a relevant environment
TRL 7	system prototype demonstration in an operational environment
TRL 8	system complete and qualified
TRL 9	actual system proven in an operational environment

Table 4

EXCLUSION CRITERIA

EC1	language other than English
EC2	full text not available
EC3	not focusing on KGs
EC4	KG not applied in a manufacturing setting
EC5	evolvability aspect not developed
EC6	poor quality
EC7	not an article (proceedings, etc.)
EC8	TRL smaller than 3

The study selection process is conducted in three stages: the search, the title and abstract screening, and the full-text screening [16]. Exclusion criteria have been created to support finding articles relevant to the review and finding research gaps. Authors have access to most academic publications through the university, but even after reasonable effort, some remained behind paywalls, which were excluded from the review. In addition, duplicate results in the search were removed from the initial set of articles.

As per the study design, the papers that did not focus on KG construction tasks or in which the KG was not developed for manufacturing were excluded. As SotA methods applied in industry are sought, a proof-of-concept (PoC) for the suggested approach is required. Thus, the study is excluded from the review if the TRL is below three (3), the levels are shown in table 3. All exclusion criteria are given in table 4.

2.3. Screening

The screening began with title and abstract screening. First, the researchers reviewed 50 studies to align their thinking and clarify necessary criteria. If it was unclear whether the study focused on manufacturing or KG construction tasks, it was moved to full-text screening if it fit other criteria. Next, the reviewer's co-reference was calculated by randomly selecting 50 studies, and all three reviewers screened them separately. The Cohen's Kappa values were above 0,57 and considered moderate [13]. Exact Kappa values from both screening phases are presented in table 5. Although the kappa coefficients were acceptable, the reviewers discussed the conflicts to align the screening process further. Then, the reviewers continued the screening in the SLR tool Covidence, and only one reviewer's opinion was required.

Table 5

REVIEWER AGREEMENT

Reviewers	Kappa phase 1	Kappa phase 2
Reviewer 1 and 2	0,57	0,73
Reviewer 1 and 3	0,60	0,71
Reviewers 2 and 3	0,77	0,72

In a full-text review, opinions from two reviewers were required for a decision, while the third reviewer solved conflicts. Based on 70 studies, the kappa coefficients were above 0.7, indicating substantial agreement. A few articles that were especially challenging to evaluate were discussed together. For instance, reviewers noticed that while many papers used link prediction for recommendations or in the initial KG development, they did not consider how the existing KG would change with link prediction. Many papers were also excluded because they recognized the need for evolvable KGs but lacked implementation.

2.4. Snowballing

Finally, forward snowballing was conducted in September 2023 based on the included articles, searching for more recent solutions. Google Scholar and Scopus were used for the snowballing to cover the majority of citers. Articles older than 03/2022 were excluded because the relevant articles published before should have been in the original review. Scopus had 384 citations of 37 articles (one article was not available in Scopus) published in 2022-2023. In the snowballing phase, title and abstract screening was conducted in the databases, after which 351 were added to full-text screening in the Covidence tool. Duplicates were removed first, and finally, 165 full texts were reviewed.

The PRISMA flowchart visualizes the whole review process in Fig. 2. Originally, 879 papers were reviewed, of which 38 were included in the review. Some were part of more extensive studies that continued in several publications, so 38 articles were included as 30 studies. After snowballing, a further eight were included in the review.

A data extraction template was used to ensure the systematic gathering of information. Two reviewers completed this template to ensure accuracy and complete information. After reviewing the data, the two reviewers agreed on the final entries. The template in table 6 included several sections aligned with the RQ 1-3. This structured approach facilitated the data collection process and supported the reliability of the information gathered.

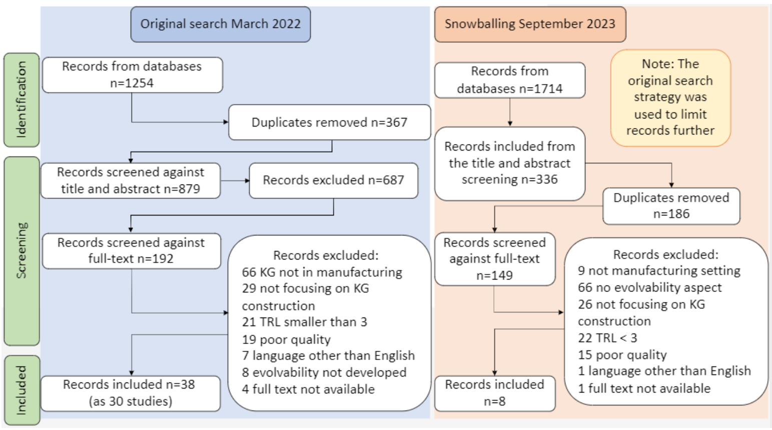

Fig. 2. Flowchart of the review process.

Table 6
DATA EXTRACTION SECTIONS

1. Authors
2. Year of publication(s)
3. Title of publication(s)
4. Aim of the study
5. Industrial setting
6. Application of KG
7. The solution utilizing KG
8. TRL
9. Automation methods for each process stage
10. Methods to change the KG

3. Literature

This section summarizes the themes found in the included articles, showcasing variations in methodologies for KG construction and applications of KGs in manufacturing. This analysis reflects on shifts in research focus and technological advancements over the past decade.

Although the search was not limited based on the time of publication, KGs have been more widely used since 2012, when Google published its Knowledge Graph. The first papers describing some aspects of evolvable KGs were published in 2017, as shown in Fig. 3. These papers represent knowledge of automated factories with KGs, and digital twins were used to represent their physical counterparts digitally. In such an environment, knowledge completion was essential to find missing information. Most relevant papers were published after 2019, following general advancements in KGs and cognitive manufacturing [65].

The papers describe various tasks to create the KG, as shown in Fig. 4. Unsurprisingly, the most described task is knowledge extraction, where entities and relations in KGs are extracted from the data, underpinning the efficiency and accuracy of the entire KG by ensuring they are accurately identified and represented. The data acquisition task includes all data preprocessing subtasks. For instance, [35] describes how data could be anonymized in manufacturing for decision support beyond the immediate environment.

Inferencing is well researched and showcased, but changes through use are less considered. Focus on applying KGs to specific cases is detectable in this table through the dominating number of studies concentrating on ontology construction, knowledge extraction, and related tasks, while cleaning and quality assessment have gained less interest. Still, their importance has been recognized by describing them as part of the process even when they have not been implemented [63, 73, 77].

Commonly, KG development is described either in a bottom-up or top-down manner, but new hybrid methods are also suggested [3, 41, 51]. The top-down method involves more human experts to create an ontology, which always requires more time, which experts might not have. The bottom-up method relies on machine learning methods to process vast amounts of data quickly, but the quality or relevance of the ontology may not be what is desired. With the hybrid techniques, humans are utilized to guarantee the quality of the KG, whereas machine learning methods offer quick analysis and initial options for humans.

Most papers describe applications on technical readiness levels 4 and 5, as shown in Fig. 5. The biased cap in scientific articles is towards these levels, whereas prototype demonstrations are primarily limited to companies and industry literature [82]. Detailed methods to construct and utilize KGs in manufacturing companies may not be published because they bring a competitive advantage to these companies. However, it is also possible that the evolvable KGs are yet to come, and these solutions represent the early stages of development.

A recurring theme in KG literature is the role of KGs in cognitive solutions. As cognitive manufacturing rises, new solutions emerge to utilize the massive data generated in manufacturing systems. For instance, cognitive twins

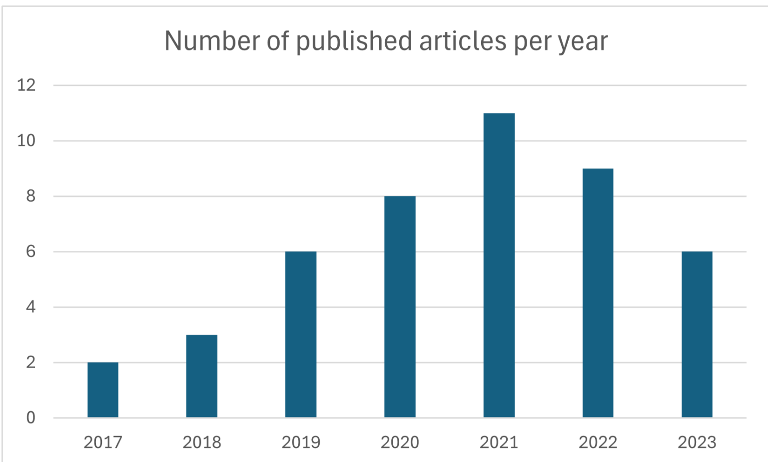

Fig. 3. Most articles were published in 2021.

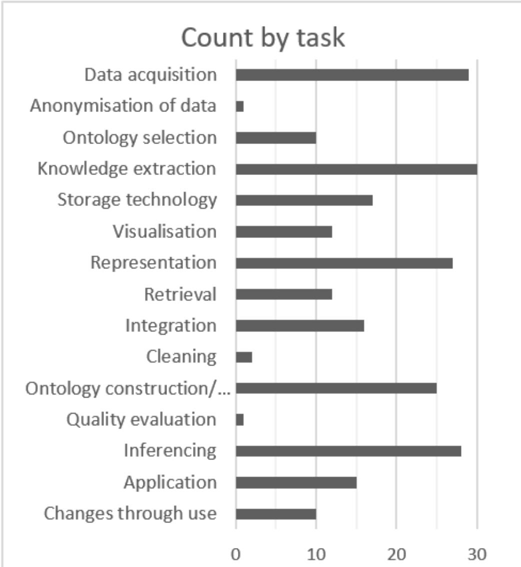

Fig. 4. Most papers still focus on data and knowledge acquisition.

are suggested as a new generation of digital twins [63], where the KG represents the knowledge acquired through the data from the physical twin and optional external sources. On the other hand, cognition is also used to enhance the KG [47]. The study describes two levels of cognition, shallow and deep, which guide and are guided by the KG.

Layered models of KGs are often used in complex environments [43, 47, 51, 73], where separate ontologies are created for different aspects of the context. The ontologies relate to each other based on class relations, such as task, product, and capability models in production, where products are required for tasks and assets have capabilities [51].

Categorization of different types of knowledge may assist in deciding how knowledge should be represented and what their applications are [30, 48]. Knowledge is classified into two dimensions: 1) structure, behavior, and function, and 2) design, monitoring, machining, control, and scheduling. Furthermore, knowledge can be classified based on the purpose, such as problem domain, solution, or evaluation knowledge [30]. Yet another classification scheme is to classify it based on what question it answers: why, what, how, or with what [29].

Evaluating the effectiveness of KGs is a challenging task requiring further research. Metrics used to evaluate them, such as sparseness or connectedness, are difficult to assess across contexts, even when using the same construction method. KGs can be evaluated from a knowledge representation perspective [10], with graph metrics [63], individual algorithms to construct the KG [28] or metrics based on the utilization environment [29, 47, 78].

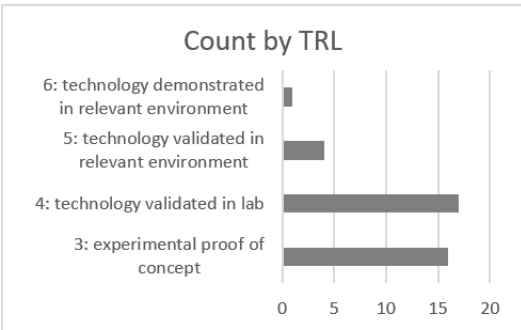

Fig. 5. TRL is relatively low in the publications.

Some solutions use virtual KGs where physical storage is in a relational database [37]. Still, it's worth noting that graph native stores are becoming increasingly popular as graph databases are used more and computer processing power allows efficient graph data processing.

KGs can also be created based on time [43]. Here, historical KGs are saved so that task progress can be searched and utilized for resource allocation in the future. Instead of triples (s, p, o), the study uses quadruples (s, p, o, t) where t is the timestamp.

3.1. Ontologies

Ontologies play a critical role in KGs as they define the schema or structure of data instances. While some sources consider ontologies to be both the structure and data instances [55], they are generally regarded as the schema for data instances [23]. In this study, KGs consist of ontologies as a schema, data instances, and possible inference engines.

The knowledge acquisition stage includes selecting an ontology or considering industrial standards. Reference ontologies or core ontologies may especially be reused. These ontologies have been designed to include the most basic concepts of a context [10]. In an industrial setting, different standards serve as a source of knowledge and guiding principles that can be employed to build ontologies [15, 48, 50]. Ontologies may be developed from standards in other information formats, such as OPC-UA [6], STEP [25, 42], QIF [42] and CAD/CAM [25]. These ontology construction methods are usually semi-automatic, as it is sufficient to specify the semantic structure for the existing structure rather than instances. Fully automatic construction has also been tried, but it is difficult [42].

Bottom-up approaches create KGs by analyzing data rather than first making the structure. For instance, ontologies can be created through text analysis [18]. An ontology construction approach based on systems thinking was introduced by [63] to focus the ontology on a specific system. Their method considers interrelations between components and forms the ontology through use case analysis. [37] found that involving field experts in ontology building was necessary. It is often mentioned that creating an ontology takes time and requires several iterations to develop it properly [22, 63]. These different aspects should be considered when developing an ontology and evaluating the best approach for the context.

Fully manual construction of the ontology is possible in well-defined, relatively simple contexts [15, 22], when it still eases the more cumbersome knowledge extraction.

3.2. Knowledge acquisition

Knowledge acquisition is the most fundamental stage for KG construction, as knowledge instances and relations are established during this stage. Perhaps because of this, this is also the most automated stage. Different ML methods are used to extract knowledge from text, documents, images, and other data formats from sensors.

When the ontology has been created, the knowledge extraction from existing databases into graphs is easier. Some methods include creating specific queries for each data source [10] but others use ML methods to achieve this. The ontology-based methods for automatic knowledge extraction are popular [10, 15, 35], as they allow domain-specific knowledge extraction. These methods require varying degrees of manual effort in creating the ontology but decrease manual work effectively.

Although automated data acquisition and knowledge extraction are well established, often efforts for domain or context-specific manipulation are required, such as enriching language models with manufacturing vocabulary [18, 60, 77]. Some algorithms may take context information as one source so that no manipulation of the algorithm is needed [1]. Sometimes, data is first acquired from text with regular expression [66], before forming a graph. It is also possible to extract graph data, in the form of triples, straight from documents, either utilizing standard description [20] or language models [60]. Here, it is enough to state rules for extracting triples instead of checking individual data. Neural networks have also emerged in this area [26].

ML methods can additionally be used to calculate which instances should have relations, [1, 72]. Many other ML methods are also used [41]. When agent-based learning is used, long-short-term memory (LSTM) is often used to enhance the agent's learning [24].

3.3. Knowledge fusion

An earlier study [69] noticed that although knowledge fusion is discussed in the KGCP, the tasks are not well defined. Here, these tasks are clarified based on the literature. Fusion deals with integrating knowledge which often introduces duplicate knowledge instances, which must be dealt with. Therefore, ontology matching and instance matching are part of this stage.

Firstly, ontology matching, or instance matching, is needed to fuse knowledge from different databases and different aspects of the industry. Ontology matching has a long history but is still a challenging task. New methods have been developed for this with ML. For instance, probabilistic soft logic (PSL) uses rules and uncertain graphs to maximize the certainty of the overall graph [20]. Ontologies are also leveraged for knowledge fusion through aligning data while extracting it from different sources [48, 60, 61]. The matching is more effortless if industrial standards and core ontologies are used [42].

Manual matching or alignment is used in well-defined cases [10, 22], but it becomes impossible for complex cases in the industry. Semantic modeling can be used to enable integration; the structure still relies heavily on manual work, but further automation work is planned [60].

Information on different manufacturing stages is often represented with distinct standards, which requires semantic alignment of the standards. [42] first create separate KGs for each stage and then fuse the KGs. The data translation from standards into KGs is done automatically, but the KG matching must be done manually.

Entity alignment is an often-used method to connect entities through similarity analysis based on structure, attributes, or textual identification, making it semiautomatic or automatic [26, 44, 47, 56, 73, 77, 79]. Entity neighborhoods can be used to reduce the computational overhead [79]. Semantic mapping on graph embeddings using the cosine theorem can also be used to infer similarity between entities [25]. Then again, [48] uses crowdsourcing to confirm integrated data if a similarity calculation is below a certain threshold.

3.3.1. Cleaning

It is important to distinguish between cleaning activities in the knowledge fusion and processing stages. Cleaning refers to checking knowledge collected with automated knowledge extraction methods. Due to integration, there is always a possibility for duplicate, irrelevant, and erroneous data when automated methods are used. In knowledge processing, quality assessment assures that the facts stored in the KG are correct.

For example, in knowledge fusion, practitioners might deal with the following type of error: machine_A has_subPart 40 Celsius, which is obviously an error because the temperature cannot be a subpart. But in knowledge processing, practitioners would deal with the following type of error: machine_A has_temperature 40 Celsius. This sounds correct but the error could be that machine A does not have a meter for temperature.

3.4. Knowledge storage

Knowledge in graphs is usually represented with the Resource Description Framework (RDF), RDF Schema, and Web Ontology Language (OWL), as these have become standards through semantic web [42]. For instance, Apache Jena is used in many papers as it is considered one of the leading RDF databases [3, 6, 50, 75].

Also, graph native options are used, like property graphs/labeled property graphs, especially Neo4j [12, 63, 64, 66, 77]. For an introduction to property graphs/labeled property graphs, see [70]. Some use a mixed approach, such as mapping RDF data to a graph database [18, 25].

Virtual storages are also used where data is in different formats in different databases (e.g. relational), but data is integrated and visualized in a graph database [10, 22]. [2] creates its own platform to enable access to multimodal data storage through the KG.

For the ontology, Protégé is certainly the most used tool [10, 25, 30, 35, 42, 44, 75], which includes rule-based inferencing algorithms.

Visualization and retrieval have not been discussed much in the literature, indicating that standard techniques available within storage technologies are used. However, Graphviz is used to visualize the graph and relations in [30].

3.5. Knowledge processing

This stage is about creating the structure for the KG and ensuring the good quality of the information stored in it. If the bottom-up approach is used, this is where the ontology is constructed through a combination of structural data analysis and expert interviews.

Quality of knowledge is an important aspect of KGs, as they are used to inform humans. It is often recognized as part of the KGCP even though it is not developed or explained in detail [47]. Quality of data has different perspectives, like representation and contextual aspects, which can also be considered in KGs [74].

In manufacturing, it seems that humans are often needed for quality checks, as [18] describes a cooperation game used for KG completion and quality check. Reinforcement learning is used for quality evaluation along with reasoning [28]. The quality of the KG also depends on only relevant information being included, so [21] removes unreachable nodes with ML methods and [28] finds wrong triples with reinforcement learning.

The next task in knowledge processing is inferencing. As inferencing has already been possible in the semantic web with rule-based methods, they are described as one part of construction in many papers. [50] claims that the rule-based and logical inferencing methods are really data preprocessing as they only add relations that should already be there, whereas ML methods may find completely new relations that are not in the structure of the ontology or its rules. However, even a simple type of reasoning adds new knowledge and is described as one type of reasoning along with representation and neural networks in [11]. The methods have increased along with computing power, and these are discussed more in the section 3.7.

Graph embedding is one of the tools that are needed in many inferencing methods, and a variety of methods are available [71]. The basic idea is that graph nodes are transformed into a low-dimensional vector space to ease computation. Methods used in the reviewed articles are discussed in section 3.7.1.

3.6. Knowledge utilization

Knowledge utilization, or knowledge consumption, in manufacturing leverages structured insights to enhance operation and support decision-making. This section explores how KGs are employed across manufacturing processes.

The most common use cases for evolvable KGs are recommender systems and information retrieval, as in Fig. 6. KGs efficiently manage massive amounts of heterogeneous data in industrial settings and question-answering. Use cases specific to manufacturing include resource allocation, diagnosis, and automation. However, evolvable KGs should consider how their utilization affects them. The active employment of KGs in manufacturing inevitably leads to their evolution. [63] recognizes the need to understand changes in the system while in use. The study utilizes systems thinking to solve challenges in building a complex system of DTs and users.

Often these two aspects overlap. For instance, [5] uses reinforcement learning for question-answering because Open Platform Communications Unified Architecture (OPC UA)-based KGs are inherently incomplete. Many other papers use link prediction to generate recommendations; these links can also enhance the KG. Furthermore, as more individualized products are manufactured, the automated production systems must be guided by past knowledge. For instance, [47] utilizes KGs to extract production requirements for new products based on semantic similarities to known products.

Different application scenarios require various tools to access and utilize the KG. Virtual reality (VR) glasses can be used in human-robot collaboration [43] or a more simple interface where humans can “discuss” with the tool to arrive at the root cause [3].

Still, most often, text-based search is used, but even in this case, the KG can offer more substantial information because of the related concepts [10]. Moreover, visual explanation can enhance the understanding of the user [1, 2].

3.7. Learning approaches in the KG

Dynamic KGs are extremely important in manufacturing due to the reliance on sensor data. This requires planning ahead for how and when updates should be executed, and is discussed in many papers [64]. As important as dynamic KGs are in manufacturing, this study explores how learning happens in the KG, which is the key difference between dynamic KG and evolvable KG. Dynamic KGs focus on updating existing data without modifying the underlying

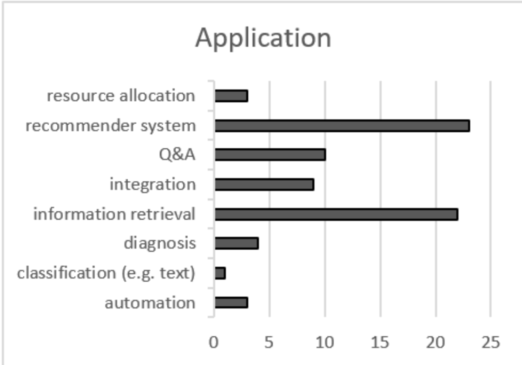

Fig. 6. There are several application scenarios for KGs in manufacturing.

structure or learning new patterns. Whereas evolvable KGs change based on learning something new in the data or about the environment it represents. The following sections explore how learning is already achieved through ML methods inside the KG and through utilizing KGs in the environment.

3.7.1. Changes through inferencing algorithms

Firstly, the evolvable KG is described in [68], where the design objective for learning inside the KG is set as follows.

> For a data analyst to allow learning through knowledge processing for KG in the intelligent industrial maintenance assistant system, employ inferencing algorithms involving data analysts because it enables finding hidden information in data.

There are 30 papers describing some form of inferencing inside the KG, which is the majority of the papers. Out of these papers, 26 use link prediction, listed in table 7, and only seven use node change, listed in table 8. Two papers were unclear on the items changed or added with the methods. Some papers are also able to discover attributes along with relations or nodes.

Graph embeddings ease the reasoning of graph information, and there are several algorithms to choose from for different situations, such as attention-based [60], adding self-loops [73], and considering the neighborhood [32, 79]. Graph neural networks (GNNs) and graph convolutional networks (GCNs) are used not only for creating embeddings, but also to learn new information [77, 79]. One paper tested four different types of embedding algorithms [5], which could be utilized for other studies. One paper created a method that uses reinforcement learning to find new information, but at the same time, it creates embeddings that can be used for further analysis [12]. There are also several papers that use word or document embedding algorithms [25, 73].

A few papers were unclear on what was added to the KG. One used ML rule-based reasoning [47] and the other embedding-based reinforcement learning [77]. The next sections will take a closer look at the relation and node changes.

Relation Changes Link prediction or knowledge reasoning has previously been categorized into three categories: distance-based, semantic-based, and embedding-based [73] or alternatively into path ranking-based, representation learning-based, and reinforcement learning-based [76]. However, the findings suggest that these methods have become increasingly diverse and are combined, making it harder to divide them into simple categories. Most notably, embeddings are achieved in different ways, and many analysis methods are used on top of them, such as mixing embeddings with reinforcement learning and so on.

A two-level categorization system reflects this complexity better. The primary basis of the reasoning is on the first level (the left column), and additional methods are on the second level (the second column). Representation-based methods have become predominant, incorporating both rule-based and reinforcement learning-based techniques.

Although tensor factorization is different from graph embedding [32], both fall under the umbrella of representation-based methods, so for simplification, both are included in the same category. In one example of the use of representation-based methods, embeddings are used to update the KG by calculating relationships based on a subgraph's internal and neighboring nodes [43]. Here, only the most significant edge weights are retained. In addition to this attention-based approach, node weights are refined using a DistMult score function to select the most likely edges [73]. Concurrently, [60] introduces an attention-based graph embedding model (ABGE) that uses a biased random walking sampling strategy along with multiple nonlinear function modules to uncover and strengthen hidden relationships within the graph.

The enhancement of KGs often involves representation-based methods coupled with similarity or distance analysis. For instance, [32] employs a dynamic scoring system, updating interactions based on item relevance to enhance accuracy.

[62] integrates event sequences into the reasoning process by preserving the temporal order of node embeddings, employing methods such as vector concatenation and recurrent neural networks (RNNs) to adapt to the event contexts. Furthermore, [80] utilizes deep learning for temporal data modeling, and then association learning for link prediction. Additionally, while primarily focusing on ontology rule-based reasoning, [78] incorporates distance analysis on node embeddings to detect implicit relations. It is likely that some type of similarity or distance analysis was also used in [53] and [43], although the studies did not specify it, because these are often incorporated within the embedding method. Link prediction is used to find answers to queries, but it is unclear if the links are added to the KG [79].

Unlike previous methods, [21] tries four different binary classification algorithms on embeddings to predict whether a link should be present or not. In [5], a reasoning agent travels on embedding vectors employing Bayesian optimization to determine optimal training parameters and uses a 3-layer LSTM. This agent is capable of uncovering new relationships in the graph it navigates, even those not explicitly encoded. Similarly, an agent travels in the graph (not vector) to find new relations to evolve the KG [28]. Also [12] describes a RL method, where agents travel in the graph, storing paths in an LSTM-based RNN. This stored knowledge is shared among agents. In another study, two RL agents explore paths from opposite directions, share paths, and update the action selection policy based on this prior knowledge [76]. However, in this paper, experts are used to verify the new relations.

Ontology-based rules have been studied long, and reasoning methods are available in most ontology tools, so there are several studies that utilize these options. In addition, ML methods can be used to classify and find rules based on the structure of the graph, like classification and regression (CART) algorithm [41], or association rule mining under incomplete evidence (AMIE) [24]. Also, expert knowledge can be utilized to define rules that describe the data and find relations based on that knowledge [1, 51]. When ML methods are used, timing and effect of processing should also be considered through creating scheduled scripts to collect changing variables [50], or setting a workflow that checks changes and triggers processing [45].

Node Changes The process of node change is more challenging, with limited options available, reflecting its inherent complexity. Despite this, similar inferencing methods, such as embedding-based and rule-based methods, are used just like in the relation change. However, the biggest difference is that human validation is sometimes used [25, 75], which is not typically involved in the relation change.

Table 7
METHODS FOR CHANGING RELATIONS

Representation (GCN, tensor factorization, node2vec, BERT etc.)	Similarity and distance analysis	[32, 60, 62, 73, 78–80]
	Reinforcement learning	[5]
	Classification	[21]
	Unclear analysis	[43, 53]
Reinforcement learning	LSTM	[12, 28]
	Human validation	[76]
Rule-based	ML rules	[24, 41]
	Ontology rules	[6, 10, 26, 30, 44, 45, 50, 78]
	Manual rules	[1, 51]

Table 8

METHODS FOR CHANGING NODES

Representation-based	Similarity and relatedness	[25, 26, 29]
	Unclear analysis	[28]
Rule-based	ML rule-based	[47, 75]
	Logical rule-based	[20]

The purpose of the change is also distinguishable. In several cases, nodes are added to construct or expand the ontology [25, 26, 29, 75]. This involves identifying and integrating ontology nodes based on analyses of text and document structures, forming an automatic ontology development method.

In terms of technical methods, GNNs are applied to generate embeddings, as seen in [28], although the specifics of the analysis remain unclear. [47] utilizes rules from quality assessment and named entity recognition to extract new information from orders, demonstrating the integration of structured rules to enhance the KG.

Furthermore, PSL is used in [20] for reasoning within uncertain KGs, where different perspectives of Cyber-Physical Systems are captured. Nodes are added based on the importance (weight) of integrated perspectives, showcasing the application of probabilistic reasoning to manage uncertainty in KGs.

3.7.2. Changes through utilizing the KG

The objective here is to allow learning, i.e., changes in the KG, while it is used through data updates or feedback, based on the following design objective.

> For a software developer and a data analyst to allow learning through knowledge utilization for KG in the system of intelligent industrial maintenance assistant, employ search behavior and feedback gathering by representing humans in the KG, involving service specialists/engineers and data analysts because doing so will enable tacit knowledge gathering from maintenance engineers and maintain timely knowledge in the KG.

Four categories were found among the learning approaches: human input, cooperation, data or knowledge updates, and machine-enriched updates. 15 studies covered learning through utilization. While many solutions hold potential, they are mostly experimental PoCs or validated in a lab environment. One paper [24] mentioned knowledge utilization through feedback; however, it was not actually implemented.

Human input Human input implies that a human expert provides feedback or input to the system through utilization, and the knowledge or learning is updated accordingly. Four studies fell into this category. In these studies, the application area of the KG is a recommender system, in addition to other possible applications. While some studies utilize neural networks, graph embeddings, and automated methods [26, 73] in knowledge acquisition, processing, or utilization, humans generally add feedback manually.

For example, in [73], an engineer can give feedback about the recommended solutions, and the feedback affects the KG either negatively or positively. On the other hand, in [54], the human gives their confidence in their choice of parameterization, and algorithms are subsequently applied to assess the influence provided by the experts.

Furthermore, in [26], the system updates resources based on human feedback, but experts can also manually update the KG by adding or removing knowledge. Finally, in [66], the system both inquires the expert to enrich the KG and interacts with the expert during the recommendation generation process. The observations by the expert are fed back into the KG, which the expert can refer to as "heuristics" the following time.

Nevertheless, while human feedback is valuable for evolving such KGs, manually inputting it back into the system is time-consuming and labor-intensive [26]. Within industrial contexts, it is crucial to establish the trustworthiness of experts and the information and knowledge they provide. Moreover, in addition to technical aspects, the resulting ethical aspects must be addressed, and the usability of the recommender should be researched [54].

Table 9

LEARNING APPROACHES THROUGH UTILIZATION

Human input	Cooperation	Data/Knowledge updates	Machine-enriched updates
[25, 54, 66, 73]	[18, 48, 63, 78]	[4, 35, 64]	[30, 43, 53, 63, 77]

Cooperation Cooperation occurs when some form of cooperation or conversation occurs between humans and machines during the same phase to update the KG. If the cooperation occurs in different stages, such as a human inputting weights directly added into the KG and then a machine processing it at another point in time, then such asynchronous collaboration does not fall into the cooperation category. Four studies were considered to fall under this category, with [63] falling under both the "cooperation" and "machine-enriched updates" categories.

[18] proposes a human-machine cooperational game that focuses on enriching the ontology by utilizing human experts. Similarly, in [48], end-users initially conduct pre-verification in a crowdsourcing phase. The validated knowledge is then refined by domain experts and integrated into the core KG, facilitating ongoing enrichment and quality assurance. While this is similar to the "human input" category, the difference is that the system (machine) cooperates in the process by calculating and determining the information that can go directly to the expert, and the information that should go through the crowdsourcing stage first.

In [78], the knowledge is utilized by forming candidate device sets for specific processes. These device sets are then evaluated and optimized. Accordingly, the information resulting from both the implicit relationships and the knowledge application is fed back to the data acquisition end to dynamically update the relationships among the workshop resources. Furthermore, implicit relationships are predicted and given to the engineers for evaluation. Once they are evaluated by the human engineers, they are updated in the KG.

Finally, [63] proposes a KG modeling approach for self-improving, actionable cognitive twins. Actionable cognitive twins result from enhancing DTs with cognitive capabilities through KG and AI models. Utilizing the knowledge from the KG and simulations provides recommendations for action. Cooperation occurs mainly in the feedback module, where feedback is collected regarding actions taken by the user and the decision-making options provided by the system. Expert experience and knowledge can also be encoded as plausible explanations or rules that can be proposed to end-users. Accordingly, the feedback can increase the knowledge base, enable new knowledge processes, identify common patterns and give rankings to actions and decision-making opportunities, inform users of future scenarios, and ask for additional knowledge from the users that may be missing.

Data or Knowledge Updates Data or knowledge updates refer to simple updates in the KG based on reasoning and finding additional relationships. Changes to the data or knowledge are then reflected in the KG. Three studies fall under this category.

In [4], maintenance log data is converted into a formal KG using a thesaurus-based method. Human selections are not considered for teaching the KG. However, the SKOS Tool was used to extend concepts, or nodes, in the thesaurus-based KG. The KG was updated based on new parsed information from websites, and new concepts found through utilization brought new relations to the graph.

In [35], a simple form of data aggregation is applied to failure data to enhance industrial maintenance. The results are "injected" into the KG to update it. While humans are involved in planning updates and making sure the information is correct when creating the KG, the KG changes through simple updates by injecting new, aggregated data into it.

Furthermore, in [64], the KG is updated through use to guide the production in (near) real-time. Users have instant access to the knowledge in the KG through a human-machine interaction application, and the KG realizes knowledge edits and updates. Additionally, when a production event is triggered, the knowledge relevant to the process, including the entities, attributes, and entity relationships, is updated in the KG.

Although these updates are similar to dynamic KGs, they are based on the utilization of the knowledge rather than only changes in sensor data. For instance, in [64], a model is proposed to support the production process by dynamically monitoring and optimizing it. This model updates manufacturing knowledge, supporting dynamic knowledge learning and reusing the proposed social-production system process.

Machine-Enriched Updates Machine-enriched updates are more complex than simply updating the data or knowledge in the KG according to changes. In this category, some form of AI, ML, graph embeddings and reasoning, or self-organization is usually applied. This provides additional knowledge or learning through KG utilization, and the KG is updated based on the results. Five studies, including [63], fall under this category.

It was previously discussed that part of [63] falls under the "cooperation" category, but another part falls under the "machine-enriched updates" category. Simulations and AI models provide forecasts by learning from past data and inductively estimating future behavior. Hence, feedback is not only provided by the user but can also be automatically obtained from the models and machine decisions and then loaded into the KG to update it through use.

Furthermore, [43] proposes a temporal subgraph reasoning-based method for self-organizing human-robot collaboration between multiple agents. The KG, which includes human, robot, component, and action nodes, provides knowledge that is communicated to and utilized by humans (e.g., through VR) and robots to guide task allocation. Multi-agent task planning, which infers and forecasts human and robotic operations for collaborative task fulfillment and efficient co-working, is generated with the updating of the human-robot collaboration KG. Thus, the self-organizing multi-agent task planning is temporally updated to adapt to changes in the task stages or the spatial locations of the components. In [53], KG embeddings are used to convert the context-enriched model into a vector space and thus enable reasoning and support an intelligent decision-making process with ML in an intelligent DT architecture. The embeddings are clustered, and the results can be fed back to the context model to enrich it further by adjusting the weights of specific relations for the self-organized reconfiguration management. Moreover, the embeddings can directly be used as input for any other ML task and accordingly incorporate the knowledge modeled in the context model into the learning process.

[30] discusses knowledge-driven navigation, where product-related information and knowledge are gathered and used as the basis for further inferring and navigation, such as through clustering, for updating the KG. This presumably makes adding new nodes easier and is convenient for recommending and predicting knowledge entities in the manufacturing KG. Although such KG updates were discussed, the study did not include implementation or further details.

Finally, [77] presents a multi-agent reinforcement learning (MARL)-enabled decentralized system to self-optimize the manufacturing process and further complement the IKG. An optimized solution is acquired through MARL, and then the characteristics of the task and the solution are packaged and manually backpropagated to the IKG. Thus, the solution space can dynamically update and adapt to generate new solutions to accomplish new tasks, and optimized solutions should be dynamically updated in the existing IKG. However, the study emphasizes the challenges and limitations of the automated establishment and dynamic evolvement of the IKG, which have not been detailed in the research.

4. DISCUSSION

The transition from dynamic KGs to evolvable KGs highlights a focus shift from changing data to learning. In an evolvable KG, change may be initiated through ML processes or by humans. Therefore, evolvable KG is not only characterized by what changes in the KG triples, as defined in section 1, but how they change. Thus, our definition of an evolvable KG is a KG that uses one or more learning approaches to change the triples. This has implications for the KGCP for evolvable KGs, which are discussed in the next sections.

4.1. KG construction process

The first contribution of the review is the refinement of the previously reviewed process models of KG construction [67, 69] based on the reviewed SotA applications (Fig. 7). Ontology construction activities are placed within the knowledge acquisition stage rather than as a separate task in the knowledge processing. This emphasizes the need for data acquisition and knowledge extraction at both the schema and instance levels. Ontology updates, typically done through inferencing or other processing methods, rightly belong in the knowledge processing stage. This change is also reflected in the knowledge fusion stage, where the integration task is now divided into ontology and instance matching.

The knowledge utilization tasks are now more descriptive. Application requirements affect KG design and should therefore be considered before KG construction. User interaction should be developed based on requirements, but the effects of the interaction must also be considered, as well as the learning enabled through that interaction.

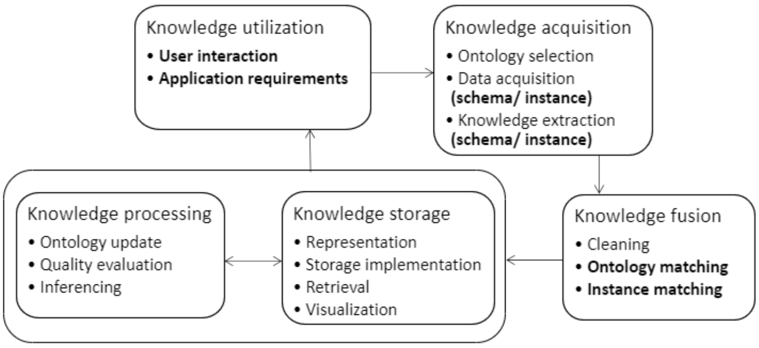

Fig. 7. The updated KG construction process where changes are highlighted with bold font.

4.2. Evolvable KG

As the second contribution, the findings enhance the theoretical understanding of evolvable KGs and distinguish them from dynamic KGs, which are characterised by changing data [43]. While the study focuses on manufacturing, the identified learning approaches likely apply across different fields. On a practical level, evolvable KGs can support Industry 4.0 and 5.0, where adaptability and effective human-machine collaboration are essential. The study has also identified the learning approaches of how KGs evolve in manufacturing settings: Different ML methods for relation and node change and changes initialized in KG utilization.

Figure 8 depicts the effects of learning approaches in the KGCP. A cyclic process is indeed needed when learning is incorporated into the process. Firstly, humans sometimes verify learning results from knowledge processing (P in Fig. 8), and may require further data processing in knowledge acquisition. The new knowledge must also be fused with existing knowledge. Similarly, utilization-based learning approaches (U in Fig. 8) are processed in knowledge acquisition and fused with existing knowledge. The knowledge discovered through utilization is typically in text or other data formats, requiring data acquisition and knowledge extraction tasks to convert this information into the appropriate format.

The learning methods through utilization are less common but sufficiently covered in the literature to allow some conclusions. The methods include information changes driven by human input, human-machine collaboration, data or knowledge updates, and enhancements through machine-driven processes. In its most basic form, this involves a human providing feedback which is then considered on its own or aggregated from multiple sources.

4.3. Challenges

Many challenges still exist in enabling learning in evolvable KGs (beyond the dynamic KGs). For instance, the papers do not discuss how often reasoning algorithms are employed. In addition, it seems that mostly just the addition of information is considered, which will become a problem in the long term. Only two studies discuss the removal of information [21, 28]. In addition, [47] adds, deletes, and modifies schema layer concepts in order to keep

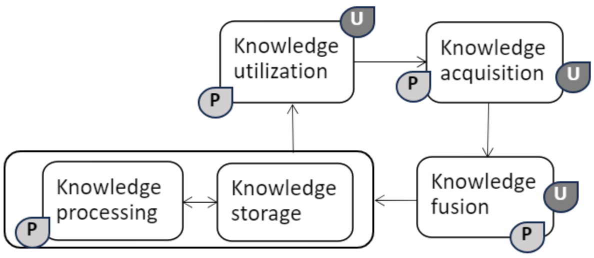

Fig. 8. Effect of learning in the process. P for ML methods and U for utilization aspects.

the KG relevant and limited. Removal of knowledge is also sometimes mentioned in utilization activities, mostly connected with human input [26].

Another issue is that fusion activities are often only considered once, while initially building the KG, but not when new information is added through ML methods or human involvement [28]. Whereas in [77], updates must go through integration to align new information with existing. Also, [78] recognizes the need to consider the knowledge acquisition phase after utilization.

As the solutions are on relatively low TRLs, many aspects of the effects of a developed solution are not considered. The ML reasoning methods are often case-dependent, so extending their use to other systems is hard. However, lessons can be learned from these cases, such as sequences of different methods and considering various data types in reasoning.

Quality evaluation becomes increasingly critical when new knowledge is added to the KG in different ways. Knowledge may also become irrelevant or inaccurate when the environment changes. Although some methods were found in the literature, many papers only mentioned (without implementation) some metrics that could be used in quality evaluation. This is problematic because it would be essential to know how these metrics are calculated and in which situations each is important. As discussed in section 3, more attention should be paid to evaluation perspectives.

The quality of knowledge within KGs significantly affects manufacturing outcomes, yet it remains an overlooked aspect. Insights from other fields could inform the approach [74]. As KGs evolve, they adjust dynamically, integrate new insights, and maintain relevance, highlighting the importance of robust quality evaluation mechanisms.

These contributions should aid in the initial implementation of evolvable KGs and their long-term maintenance and relevance. The process description and categories of learning approaches are intended to help practitioners manage the complexities of maintaining evolving KGs, addressing the integration of new knowledge, and removing outdated information.

5. CONCLUSION

This study has systematically reviewed the development and application of evolvable KGs within the manufacturing sector, highlighting their critical role in enhancing the adaptability and intelligence of industrial systems. The study has identified the SotA and prevailing gaps in KG evolution through a detailed analysis of current literature and methodologies.

The findings demonstrate that while the current state-of-the-art has focused on dynamic knowledge graphs, evolvable KGs become critical to realizing Industry 4.0 and 5.0, offering dynamic solutions that integrate seamlessly with the rapid changes and increasing complexity of modern manufacturing environments. The ability of these KGs to learn and adapt through machine learning algorithms and human expertise enhances decision-making processes, operational efficiency, and predictive maintenance.

However, despite these advancements, the review also points out the challenges related to quality assurance, process planning, and incorporating human expertise within systems. Addressing these challenges is crucial for the broader adoption and optimization of KG technologies in manufacturing.

In conclusion, evolvability and related learning approaches should be considered when designing KGs and associated systems in the industrial context – unless the knowledge structures and environments are expected to remain stable over the KG's life cycle and related applications.

References

[1] H. Abu-Rasheed, C. Weber, J. Zenkert, M. Dornhöfer and M. Fathi, Transferrable Framework Based on Knowledge Graphs for Generating Explainable Results in Domain-Specific, Intelligent Information Retrieval **9**(1). doi:10.3390/informatics9010006.

[2] K.S. Aggour, V.S. Kumar, P. Cuddihy, J.W. Williams, V. Gupta, L. Dial, T. Hanlon, J. Gambone and J. Vinciquerra, Federated Multimodal Big Data Storage & Analytics Platform for Additive Manufacturing, in: *2019 IEEE International Conference on Big Data (Big Data)*, pp. 1729–1738. doi:10.1109/BigData47090.2019.9006495.

[3] F. Ameri and R. Yoder, A Thesaurus-Guided Method for Smart Manufacturing Diagnostics. doi:10.1007/978-3-030-30000-5$_8$8.
[4] F. Ameri, R. Yoder and K. Zandbiglari, SKOS Tool: A Tool for Creating Knowledge Graphs to Support Semantic Text Classification. doi:10.1007/978-3-030-57997-5$_3$1.
[5] J. Bakakeu, F. Schäfer, J. Franke, S. Baer, H.-H. Klos and J. Peschke, Reasoning over OPC UA Information Models using Graph Embedding and Reinforcement Learning, pp. 40–47. doi:10.1109/AI4I49448.2020.00016.
[6] R.J. Bakakeu, M. Brossog, S. Bär, F. Schäfer, J. Franke, S. Tolksdorf, H.-H. Klos and J. Peschke, Automated Reasoning and Knowledge Inference on OPC UA Information Models, pp. 53–60. doi:10.1109/ICPHYS.2019.8780114.
[7] T. Berners-Lee, Linked Data. https://www.w3.org/DesignIssues/LinkedData.html.
[8] A. Breemen, J. Farkas and J. Sarbo,
Knowledge Representation as a Tool for Intelligence Augmentation. doi:10.4018/978-1-60960-551-3.ch013.
[9] G. Buchgeher, D. Gabauer, J. Martinez-Gil and L. Ehrlinger, Knowledge Graphs in Manufacturing and Production: A Systematic Literature Review
9, 55537–55554, Conference Name: IEEE Access. doi:10.1109/ACCESS.2021.3070395.
[10] Q. Cao, S. Beden and A. Beckmann, A core reference ontology for steelmaking process knowledge modelling and information management
135. doi:10.1016/j.compind.2021.103574.
[11] X. Chen, S. Jia and Y. Xiang, A review: Knowledge reasoning over knowledge graph
141. doi:10.1016/j.eswa.2019.112948.
[12] Y. Chi, Z.J. Wang and V.C.M. Leung,
Distributed Knowledge Inference Framework for Intelligent Fault Diagnosis in IIoT systems, 1–1. doi:10.1109/TNSE.2021.3128171.
[13] J. Cohen, A Coefficient of Agreement for Nominal Scales
20(1), 37–46, Publisher: SAGE Publications Inc. doi:10.1177/001316446002000104.
[14] D. Dellermann, P. Ebel, M. Söllner and J.M. Leimeister, Hybrid Intelligence
61(5), 637–643. doi:10.1007/s12599-019-00595-2.
[15] J. Dong, J. Wang and S. Chen, Knowledge graph construction based on knowledge enhanced word embedding model in manufacturing domain
41(2), 3603–3613. doi:10.3233/JIFS-210982.
[16] T. Dyba, T. Dingsoyr and G.K. Hanssen, Applying Systematic Reviews to Diverse Study Types: An Experience Report, in: *First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007)*, IEEE, pp. 225–234. ISBN 978-0-7695-2886-1. doi:10.1109/ESEM.2007.59.
[17] N. Elgendy, Enhancing Collaborative Rationality between Humans and Machines through Data-Driven Decision Evaluation, CEUR.
[18] E. Fenoglio, E. Kazim, H. Latapie and A. Koshiyama, Tacit knowledge elicitation process for industry 4.0
2(1), 6. doi:10.1007/s44163-022-00020-w.
[19] I. Grangel-González and M.E. Vidal, Analyzing a Knowledge Graph of Industry 4.0 Standards, in: *The Web Conference 2021 - Companion of the World Wide Web Conference, WWW 2021*, pp. 16–25. doi:10.1145/3442442.3453542.
[20] I. Grangel-González, L. Halilaj, M.-E. Vidal, O. Rana, S. Lohmann, S. Auer and A.W. Müller, *Knowledge Graphs for Semantically Integrating Cyber-Physical Systems*, Vol. 11029 LNCS, Pages: 199. doi:10.1007/978-3-319-98809-2$_1$2.
[21] I. Grangel-González and F. Shah, Link Prediction with Supervised Learning on an Industry 4.0 related Knowledge Graph, in: *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1–8. doi:10.1109/ETFA45728.2021.9613314.
[22] I. Grangel-González, F. Lösch and A. ul Mehdi, Knowledge Graphs for Efficient Integration and Access of Manufacturing Data, in: *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Vol. 1, pp. 93–100, ISSN: 1946-0759. doi:10.1109/ETFA46521.2020.9212156.
[23] T.R. Gruber, A Translation Approach to Portable Ontology Specifications - Definition of Ontology as Specification
5(2), 199–220. https://tomgruber.org/writing/ontolingua-kaj-1993.
[24] K. Guan, Y. Sun, G. Yang and X. Yang, Knowledge Acquisition and Reasoning Model for Welding Information Integration Based on CNN and Knowledge Graph
12(6), 1275, Number: 6 Publisher: Multidisciplinary Digital Publishing Institute. doi:10.3390/electronics12061275. https://www.mdpi.com/2079-9292/12/6/1275.
[25] L. Guo, F. Yan, Y. Lu, M. Zhou and T. Yang, An automatic machining process decision-making system based on knowledge graph
34(12), 1348–1369. doi:10.1080/0951192X.2021.1972461.
[26] L. Guo, F. Yan, T. Li, T. Yang and Y. Lu, An automatic method for constructing machining process knowledge base from knowledge graph
73. doi:10.1016/j.rcim.2021.102222.
[27] J.M. Gómez-Berbís and A. Amescua-Seco, *Sedit: Semantic digital twin based on industrial iot data management and knowledge graphs*, Vol. 1124 CCIS, Pages: 188. doi:10.1007/978-3-030-34989-9$_1$4.
[28] H. Han, J. Wang, X. Wang and S. Chen, Construction and Evolution of Fault Diagnosis Knowledge Graph in Industrial Process
71, 1–12, Conference Name: IEEE Transactions on Instrumentation and Measurement. doi:10.1109/TIM.2022.3200429. https://ieeexplore.ieee.org/abstract/document/9863829.
[29] L. He, B. Dong and P. Jiang, A Heuristic Grafting Strategy for Manufacturing Knowledge Graph Extending and Completion Based on Nature Language Processing: KnowTree
9, 90847–90862. doi:10.1109/ACCESS.2021.3092019.

[30] L. He and P. Jiang, Manufacturing Knowledge Graph: A Connectivism to Answer Production Problems Query With Knowledge Reuse **7**, 101231–101244, Conference Name: IEEE Access. doi:10.1109/ACCESS.2019.2931361.

[31] T. Heath and C. Bizer, *Linked Data: Evolving the Web into a Global Data Space*, *Synthesis Lectures on Data, Semantics, and Knowledge*, Springer International Publishing. ISBN 978-3-031-79431-5 978-3-031-79432-2. doi:10.1007/978-3-031-79432-2.

[32] M. Hildebrandt, S.S. Sunder, S. Mogoreanu, I. Thon, V. Tresp and T. Runkler, Configuration of industrial automation solutions using multi-relational recommender systems, in: *Lecture Notes in Artificial Intelligence*, Vol. 11053, Springer Verlag, pp. 271–287. doi:10.1007/978-3-030-10997-4$_1$7.

[33] P. Hitzler,
Semantic Web: A Review Of The Field.

[34] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutierrez, J.E.L. Gayo, S. Kirrane, S. Neumaier, A. Polleres, R. Navigli, A.-C.N. Ngomo, S.M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab and A. Zimmermann, Knowledge Graphs **54**(4), 1–37. doi:10.1145/3447772. http://arxiv.org/abs/2003.02320.

[35] H. Hossayni, I. Khan, M. Aazam, A. Taleghani-Isfahani and N. Crespi, SemKoRe: Improving Machine Maintenance in Industrial IoT with Semantic Knowledge Graphs
10(18), 6325, Number: 18 Publisher: Multidisciplinary Digital Publishing Institute. doi:10.3390/app10186325. https://www.mdpi.com/2076-3417/10/18/6325.

[36] E. Hyvönen, J. Tuominen, M. Alonen and E. Mäkelä, Linked Data Finland: A 7-star Model and Platform for Publishing and Re-using Linked Datasets, in: *The Semantic Web: ESWC 2014 Satellite Events*, V. Presutti, E. Blomqvist, R. Troncy, H. Sack, I. Papadakis and A. Tordai, eds, Lecture Notes in Computer Science, Springer International Publishing, pp. 226–230. ISBN 978-3-319-11955-7. doi:10.1007/978-3-319-11955-7$_2$4.

[37] E.G. Kalaycı, I. Grangel González, F. Lösch, G. Xiao, A. ul-Mehdi, E. Kharlamov and D. Calvanese, Semantic Integration of Bosch Manufacturing Data Using Virtual Knowledge Graphs, in: *The Semantic Web – ISWC 2020*, J.Z. Pan, V. Tamma, C. d'Amato, K. Janowicz, B. Fu, A. Polleres, O. Seneviratne and L. Kagal, eds, Lecture Notes in Computer Science, Springer International Publishing, pp. 464–481. ISBN 978-3-030-62466-8. doi:10.1007/978-3-030-62466-8$_2$9.

[38] M.H. Karray, F. Ameri, M. Hodkiewicz and T. Louge, ROMAIN: Towards a BFO compliant reference ontology for industrial maintenance **14**(2), 155–177, Publisher: IOS Press. doi:10.3233/AO-190208. https://content.iospress.com/articles/applied-ontology/ao190208.

[39] B. Kitchenham and P. Brereton, A systematic review of systematic review process research in software engineering
55(12), 2049–2075. doi:10.1016/j.infsof.2013.07.010. https://www.sciencedirect.com/science/article/pii/S0950584913001560.

[40] B.A. Kitchenham, Procedures for Performing Systematic Reviews, Library Catalog: www.semanticscholar.org. /paper/Procedures-for-Performing-Systematic-Reviews-Kitchenham/29890a936639862f45cb9a987dd599dce9759bf5.

[41] H. Ko, P. Witherell, Y. Lu, S. Kim and D.W. Rosen, Machine learning and knowledge graph based design rule construction for additive manufacturing
37. doi:10.1016/j.addma.2020.101620.

[42] S. Kwon, L.V. Monnier, R. Barbau and W.Z. Bernstein, Enriching standards-based digital thread by fusing as-designed and as-inspected data using knowledge graphs
46. doi:10.1016/j.aei.2020.101102.

[43] S. Li, P. Zheng, S. Pang, X.V. Wang and L. Wang, Self-organising multiple human–robot collaboration: A temporal subgraph reasoning-based method
68, 304–312. doi:10.1016/j.jmsy.2023.03.013. https://www.sciencedirect.com/science/article/pii/S027861252300064X.

[44] X. Li, S. Zhang, R. Huang, B. Huang, C. Xu and B. Kuang, Structural Modeling of Heterogeneous CAM Model Based on Process Knowledge Graph
30(7), 1342–1355. doi:10.3724/SP.J.1089.2018.16702.

[45] T. Liebig, A. Maisenbacher, M. Opitz, J.R. Seyler, G. Sudra and J. Wissmann, Building a knowledge graph for products and solutions in the automation industry, in: *CEUR Workshop Proceedings*, Vol. 2489, pp. 13–23.

[46] J. Liu, Q. Zhang, L. Fu, X. Wang and S. Lu, Evolving Knowledge Graphs, in: *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, IEEE, pp. 2260–2268. ISBN 978-1-72810-515-4. doi:10.1109/INFOCOM.2019.8737547. https://ieeexplore.ieee.org/document/8737547/.

[47] M. Liu, X. Li, J. Li, Y. Liu, B. Zhou and J. Bao, A knowledge graph-based data representation approach for IIoT-enabled cognitive manufacturing
51, 101515. doi:10.1016/j.aei.2021.101515. https://www.sciencedirect.com/science/article/pii/S1474034621002639.

[48] M. Lyu, X. Li and C.-H. Chen, Achieving Knowledge-as-a-Service in IIoT-driven smart manufacturing: A crowdsourcing-based continuous enrichment method for Industrial Knowledge Graph
51, 101494. doi:10.1016/j.aei.2021.101494. https://www.sciencedirect.com/science/article/pii/S1474034621002433.

[49] J. Martinez-Gil, T. Hoch, M. Pichler, B. Heinzl, B. Moser, K. Kurniawan, E. Kiesling and F. Krause, Examining the Adoption of Knowledge Graphs in the Manufacturing Industry: A Comprehensive Review, in: *Artificial Intelligence in Manufacturing: Enabling Intelligent, Flexible and Cost-Effective Production Through AI*, J. Soldatos, ed., Springer Nature Switzerland, pp. 55–70. ISBN 978-3-031-46452-2. doi:10.1007/978-3-031-46452-2$_4$.

[50] S. Meckler, H. Steinmüller and A. Harth, Building a Knowledge Graph with Inference for a Production Machine Using the Web of Things Standard, in: *Advances and Trends in Artificial Intelligence. From Theory to Practice*, H. Fujita, A. Selamat, J.C.-W. Lin and M. Ali, eds, Lecture Notes in Computer Science, Springer International Publishing, pp. 240–251. ISBN 978-3-030-79463-7. doi:10.1007/978-3-030-79463-7$_2$0.

[51] F. Mo, J.C. Chaplin, D. Sanderson, G. Martínez-Arellano and S. Ratchev, Semantic models and knowledge graphs as manufacturing system reconfiguration enablers
86, 102625. doi:10.1016/j.rcim.2023.102625. https://www.sciencedirect.com/science/article/pii/S073658452300100X.

[52] T. Mortlock, D. Muthirayan, S.-Y. Yu, P.P. Khargonekar and M.A.A. Faruque, Graph Learning for Cognitive Digital Twins in Manufacturing Systems
10(1), 34–45. doi:10.1109/TETC.2021.3132251.

[53] T. Muller, N. Sahlab, S. Kamm, C. Kohler, D. Braun, N. Jazdi and M. Weyrich, Context-enriched modeling using Knowledge Graphs for intelligent Digital Twins of Production Systems, in: *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE, pp. 1–8. ISBN 978-1-66549-996-5. doi:10.1109/ETFA52439.2022.9921615.

[54] R. Nordsieck, M. Heider, A. Winschel and J. Hahner, Knowledge Extraction via Decentralized Knowledge Graph Aggregation, in: *Proceedings - 2021 IEEE 15th International Conference on Semantic Computing, ICSC 2021*, pp. 92–99. doi:10.1109/ICSC50631.2021.00024.

[55] N.F. Noy and D.L. McGuinness,
Ontology Development 101: A Guide to Creating Your First Ontology.

[56] F. Ocker, B. Vogel-Heuser and C.J.J. Paredis, A framework for merging ontologies in the context of smart factories
135, 103571. doi:10.1016/j.compind.2021.103571. https://www.sciencedirect.com/science/article/pii/S0166361521001780.

[57] A. Polleres, R. Pernisch, A. Bonifati, D. Dell'Aglio, D. Dobriy, S. Dumbrava, L. Etcheverry, N. Ferranti, K. Hose, E. Jiménez-Ruiz, M. Lissandrini, A. Scherp, R. Tommasini and J. Wachs,
How Does Knowledge Evolve in Open Knowledge Graphs?, 59 pages, 4360705 bytes–, Artwork Size: 59 pages, 4360705 bytes Medium: application/pdf Publisher: Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/TGDK.1.1.11.

[58] M.R. Quillian, Word concepts: A theory and simulation of some basic semantic capabilities
12(5), 410–430, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/bs.3830120511. doi:10.1002/bs.3830120511.

[59] H. Ren, D. Anicic and T.A. Runkler,
Towards Semantic Management of On-Device Applications in Industrial IoT. https://doi.org/10.1145/3510820.

[60] L. Ren, Y. Li, X. Wang, J. Cui and L. Zhang,
An ABGE-aided manufacturing knowledge graph construction approach for heterogeneous IIoT data integration, 1–15, ISBN: 0020-7543 Publisher: Taylor & Francis.

[61] M. Ringsquandl, E. Kharlamov, D. Stepanova, M. Hildebrandt, S. Lamparter, R. Lepratti, I. Horrocks and P. Kröger, *Event-Enhanced Learning for KG Completion*, Vol. 10843 LNCS, Pages: 559. doi:10.1007/978-3-319-93417-4$_3$5.

[62] M. Ringsquandl, E. Kharlamov, D. Stepanova, S. Lamparter, R. Lepratti, I. Horrocks and P. Kroger, On event-driven knowledge graph completion in digital factories, in: *Proceedings - 2017 IEEE International Conference on Big Data, Big Data 2017*, Vol. 2018-January, pp. 1676–1681. doi:10.1109/BigData.2017.8258105.

[63] J.M. Rožanec, J. Lu, J. Rupnik, M. Škrjanc, D. Mladenić, B. Fortuna, X. Zheng and D. Kiritsis, Actionable cognitive twins for decision making in manufacturing
60(2), 452–478. doi:10.1080/00207543.2021.2002967.

[64] H. Shi, M. Yang and P. Jiang, Social production system: A three-layer smart framework for implementing autonomous human-machine collaborations in a shop floor
9, 26696–26711. doi:10.1109/ACCESS.2021.3050236.

[65] Stephen DeAngelis, Industry 4.0: The Emergence of Cognitive Manufacturing - Enterra Solutions. https://enterrasolutions.com/industry-4-0-the-emergence-of-cognitive-manufacturing/.

[66] S. Sunkle, D. Jain, K. Saxena, A. Patil, T. Singh, B. Rai and V. Kulkarni, Integrated "generate, make, and test" for formulated products using knowledge graphs
3(3), 340–375. doi:10.1162/dint$_a$0096.

[67] G. Tamašauskaite and P. Groth, Defining a Knowledge Graph Development Process Through a Systematic Review
32(1). doi:10.1145/3522586.

[68] A. Teern, M. Kelanti, T. Päivärinta and M. Karaila,
Design Objectives for Evolvable Knowledge Graphs, 1–15, Number: 36. doi:10.7250/csimq.2023-36.01. https://csimq-journals.rtu.lv/article/view/csimq.2023-36.01.

[69] A. Teern, M. Kelanti, T. Päivärinta and M. Karaila, Knowledge graph construction and maintenance process: Design challenges for industrial maintenance support, in: *BIR 2022 Workshops and Doctoral Consortium*, Vol. 3223, CEUR, pp. 1–13. ISBN 1613-0073. urn:nbn:de:0074-3223-0.

[70] TerminusDB, Graph Fundamentals - Part 2: Labeled Property Graphs. https://terminusdb.com/blog/graph-fundamentals-part-2-labeled-property-graphs/.

[71] Q. Wang, Z. Mao, B. Wang and L. Guo, Knowledge Graph Embedding: A Survey of Approaches and Applications
29(12), 2724–2743. doi:10.1109/TKDE.2017.2754499.

[72] Y. Wang, L. Gao, Y. Gao, X. Li and L. Gao, Knowledge Graph-guided Convolutional Neural Network for Surface Defect Recognition, in: *IEEE International Conference on Automation Science and Engineering*, Vol. 2020-August, pp. 594–599. doi:10.1109/CASE48305.2020.9216752.

[73] L. Xia, Y. Liang, J. Leng and P. Zheng, Maintenance planning recommendation of complex industrial equipment based on knowledge graph and graph neural network
232, 109068. doi:10.1016/j.ress.2022.109068.

[74] B. Xue and L. Zou,
Knowledge Graph Quality Management: a Comprehensive Survey. doi:10.1109/TKDE.2022.3150080.
[75] C. Zhang, G. Zhou, F. Chang and X. Yang, Learning domain ontologies from engineering documents for manufacturing knowledge reuse by a biologically inspired approach
106(5), 2535–2551. doi:10.1007/s00170-019-04772-1.
[76] Y. Zhang, H. Wang, W. Shen and G. Peng,
DuAK: Reinforcement Learning-Based Knowledge Graph Reasoning for Steel Surface Defect Detection, 1–13, Conference Name: IEEE Transactions on Automation Science and Engineering. doi:10.1109/TASE.2023.3307588. https://ieeexplore.ieee.org/abstract/document/10247201.
[77] P. Zheng, L. Xia, C. Li, X. Li and B. Liu, Towards Self-X cognitive manufacturing network: An industrial knowledge graph-based multi-agent reinforcement learning approach
61, 16–26. doi:10.1016/j.jmsy.2021.08.002. https://www.sciencedirect.com/science/article/pii/S0278612521001643.
[78] B. Zhou, J. Bao, J. Li, Y. Lu, T. Liu and Q. Zhang, A novel knowledge graph-based optimization approach for resource allocation in discrete manufacturing workshops
71. doi:10.1016/j.rcim.2021.102160.
[79] B. Zhou, B. Hua, X. Gu, Y. Lu, T. Peng, Y. Zheng, X. Shen and J. Bao, An end-to-end tabular information-oriented causality event evolutionary knowledge graph for manufacturing documents
50, 101441. doi:10.1016/j.aei.2021.101441. https://www.sciencedirect.com/science/article/pii/S1474034621001932.
[80] B. Zhou, X. Shen, Y. Lu, X. Li, B. Hua, T. Liu and J. Bao, Semantic-aware event link reasoning over industrial knowledge graph embedding time series data
0(0), 1–18, Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/00207543.2021.2022803. doi:10.1080/00207543.2021.2022803.
[81] X. Zou, A Survey on Application of Knowledge Graph
1487, 012016, Publisher: IOP Publishing. doi:10.1088/1742-6596/1487/1/012016.
[82] Guiding notes to use the TRL self-assessment tool, BRIDGE2HE project, APRE = Agency for the Promotion of European Research, CDTI = Centre for the Development of Industrial Technology.