A knowledge graph approach for state-of-the-art implementation of industrial factory movement tracking system

Citation for published version:
Vasantha, G, Aslan, A, Hanson, J, El-Raoui, H, Corney, J & Quigley, J 2023, A knowledge graph approach for state-of-the-art implementation of industrial factory movement tracking system. in Flexible Automation and Intelligent Manufacturing: Establishing Bridges for More Sustainable Manufacturing Systems: Proceedings of FAIM 2023, June 18–22, 2023, Porto, Portugal, Volume 2: Industrial Management. vol. 2, Lecture Notes in Mechanical Engineering, Springer, Cham, pp. 1194-1204, International Conference on Flexible Automation and Intelligent Manufacturing 2023, Porto, Portugal, 18/06/23. https://doi.org/10.1007/978-3-031-38165-2_136

Digital Object Identifier (DOI):
10.1007/978-3-031-38165-2_136

Published In:
Flexible Automation and Intelligent Manufacturing: Establishing Bridges for More Sustainable Manufacturing Systems

A KNOWLEDGE GRAPH APPROACH FOR STATE-OF-THE-ART IMPLEMENTATION OF INDUSTRIAL FACTORY MOVEMENT TRACKING SYSTEM

Gokula Vasantha[1[0000-0002-5479-6134]], Ayse Aslan[1[0000-0003-2974-8314]], Jack Hanson[2[0000-0001-5105-8497]], Hanane El-Raoui[3[0000-0002-9079-3248]], Jonathan Corney[2[0000-0003-1210-3827]] and John Quigley[3[0000-0002-7253-8470]]

[1] Edinburgh Napier University, EH10 5DT, UK
[2] The University of Edinburgh, EH8 9YL, UK
[3] University of Strathclyde, G1 1XQ, UK
g.vasantha@napier.ac.uk

Abstract. Digital sensing technologies are essential for realizing Industry 4.0, as they enhance productivity, assist with real-time decision-making, and provide flexibility and agility in manufacturing factories. However, implementing these technologies can be a significant challenge due to the need to consider various factors in manufacturing factories, such as heterogeneous equipment, fragmented knowledge, customization requirements, multiple alternative technologies, and the substantial costs involved in the trial-and-error process. A Knowledge Graph (KG) approach is proposed to streamline the implementation of the factory movement tracking system. The KG approach utilizes a knowledge representation reference model that integrates manufacturing objective, activity, resource, environment, factory movement, data, infrastructure, and decision support system. This reference model aids in classifying key phrases extracted from research abstracts and establishing knowledge relationships among them. A synthesized KG, created by analyzing thirty research abstracts, has correctly answered search queries about implementing the factory movement tracking system. This approach establishes a pathway for developing a software system to support movement tracking implementation through automatic interpretation, reasoning, and suggestions.

Keywords: Factory movement, Industry implementation, Knowledge Graph, Knowledge representation.

1 Introduction

Manufacturing companies are transitioning from one technology revolution to the next to enhance productivity, agility, and flexibility in their production processes. In this context, Industry 4.0 is revolutionizing the manufacturing environment by integrating multiple new technologies, such as the Internet of Things (IoT), big data, cloud computing, analytics, and artificial intelligence, to create a smart manufacturing environment. For example, advanced sensors facilitate the collection and analysis of factory data in real time, enabling rapid decision-making, automation, process and quality improvements, improving production capacity, and minimizing downtime through efficient maintenance. In other words, the technologies have the potential to give a high level of visibility of the state of all processes across a factory.

However, implementing Industry 4.0 technology can be complex due to the need to consider various factors within a manufacturing factory, such as heterogeneous equipment, isolated and fragmented knowledge, customization requirements, multiple alternative technologies, and the significant costs involved in the trial-and-error process. These factors can make it challenging to successfully implement Industry 4.0 technology and fully realize its potential benefits. Ing et al. [1] identified seven important challenges in implementing Industry 4.0: data management and integration, knowledge-driven, process, security, capital, workforce, and education. Moktadir et al. [2] pointed out that the lack of technological infrastructure is the biggest issue in Industry 4.0 implementation. The studies within German manufacturing companies show that the competitiveness, future viability, and organizational and production fit impede the implementation of Industry 4.0 [3]. Veile et al. [4] observed that educating employees with new competencies and knowledge, necessary organizational and cultural changes, openness and trust, and integration into the existing machinery and production systems play essential roles in implementation. The challenges associated with implementing Industry 4.0 technology underscore the importance of systematically considering multiple aspects of the manufacturing factory and having up-to-date domain knowledge to overcome these challenges.

As new technologies and methods for Industry 4.0 are proposed and evaluated in the literature, it is necessary to systematically accumulate this knowledge so it can be stored, organized, and shared within organizations or external stakeholders. Therefore, the challenge is to encode and represent the knowledge reported in the literature to facilitate progression as the field matures. Since Industry 4.0 is a large subject, this research focuses on creating a Knowledge Graph (KG) approach for implementing an industrial factory movement tracking system.

The paper is organized as follows: First, it reviews existing approaches for implementing support. Then, it presents the research aim and methodology. Next, it explains the proposed knowledge representation reference model and KG generation approach, including an example. The paper then discusses the assessment and validation of the generated KG, and concludes with the findings and suggestions for future work.

2 Literature on Existing Industry 4.0 Implementation Support Approaches

The section reviews the reference architectures, observed patterns and ontologies (with any associated standards) used to support the implementation of Industry 4.0 technologies and identifies research gaps.

A **Reference architecture** is a high-level, abstract document that outlines the overall structure and organization of a system. In the context of Industry 4.0, reference architectures can provide the overall structure for systems and support the selection and integration of their hardware and software components. Multiple literature sources have reviewed the existing reference architectures [5-7]. The common reference architectures highlighted are: IIRA (Industrial Internet Reference Architecture), RAMI 4.0 (Reference Architectural Model Industrie 4.0), SITAM (Stuttgart IT-Architecture for Manufacturing), LASFA (LAsim Smart FActory), NIST Smart Manufacturing architecture, and IBM Industry 4.0. These Industry 4.0 reference architectures are primarily represented in layers/levels, building blocks, and communication among them.

These architectures cover broad topics of smart factories from the business to the shop floor levels, such as business structure, operation, prognostics, optimization, information analytics, and monitoring/control of devices. Commonly these architectures were compared across the following five levels of automation architecture: (i) field (represents physical entities on the production floor); (ii) control (control a physical entity primarily using a PLC (Programmable Logic Controller) and a PID (Proportional–Integral–Derivative controller); (iii) system/process (controls multiple PLCs, e.g. SCADA (Supervisory Control and Data Acquisition)); (iv) operation (systems that monitor the entire manufacturing process - MES (Manufacturing Execution System)); and (v) enterprise (systems for integrated management; ERP (Enterprise Resource Planning)). Although the reference architectures tend to increase interoperability among systems/subsystems, reduce development costs/time, enable effective knowledge reuse, and adopt best practices, the following challenges in utilizing reference architectures are observed:

- High abstraction leads to difficulty in instantiating them to real-world Industry 4.0 projects.
- Difficulties in mapping terminologies used across reference architectures for representing similar concepts.
- The literature on architecture implementation lacks detailedness, particularly in analyzing architecture's internal structure, components, communication, data exchanges and types, and decision-making capabilities.
- Updating the evolving nature of the scope, technologies and systems involved is challenging.
- Drawbacks to apply in customized use cases for varying industrial requirements.

IoT patterns can be understood as a collection of problems and solutions within specific contexts. Bloom et al. [8] identify common input-output design patterns to understand data flow semantics in IoT applications: Closed-Loop, Cloud-in-the-Loop, Open-Loop, Cloud-on-the-Loop, and Device-to-Device. Washizaki et al. [9] observed 61 IoT design patterns such as 'Entity-Component-Attribute', 'Actuation-Actuator-Effect', 'Operator-Controller-Module' in published research articles. They noted that these patterns are not referenced frequently except 'Operator-Controller-Module'. These patterns are not well classified, varied in abstraction levels, and their application adoption is low.

Several **ontologies** have been proposed to describe resources, processes and location navigation. Some of the established ontologies are: Semantic Sensor Network (SSN) ontology [10], manufacturing resources and integration - IEC / ISO 62264 [11], and Process Specification Language (PSL) [12]. These ontologies provide detailed specifications in a particular domain by establishing terminologies and relationship definitions. Although these ontologies are well defined, integrating these ontologies to a common purpose of structuring knowledge progression through published literature is not created yet. This paper aims to address these limitations observed by developing a Knowledge Graph (KG) based approach for systematically learning technology progression in factory movement tracking systems. This approach should support and drive the implementation of Industry 4.0 systems for realizing smart factories.

3 Research Aim and Methodology

This research aims to develop a support tool that systematically captures the knowledge generated in implementing Industry 4.0 processes and technologies. Structuring and reusing learnt knowledge will facilitate effective and streamlined industrial implementation and prevent costly trial-and-error procedures. This paper presents a Knowledge Graph (KG) approach for organizing and structuring abstracts from published literature about implementing a factory movement tracking system. This approach requires both a knowledge representation reference model and a process for creating the KG from the abstracts of research papers. Figure 1 details the process of creating and assessing a KG. Thirty research papers have been chosen based on the keyword search term 'factory movement data' in Google Scholar and analyzed using the described steps. Studying the first 30 relevant articles provided sufficient breadth and depth to validate the proposed KG approach for the movement tracking system. Protégé software was used to input the structured KG and assess it through SPARQL search queries. The following sections detail the steps mentioned in Figure 1.

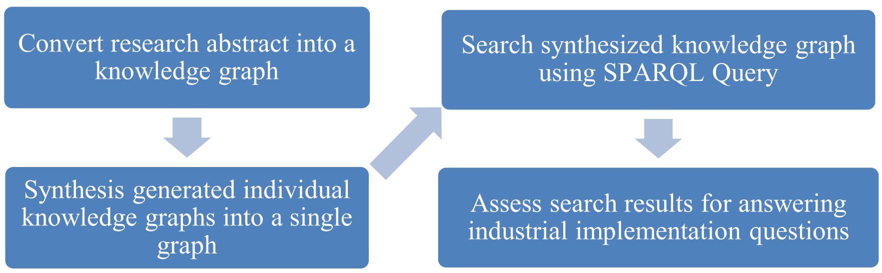

Fig. 1. Methodology to generate and assess synthesized Knowledge Graph

4 Knowledge Representation Reference Model

The proposed knowledge representation reference model provides overarching framework elements to consider when implementing factory movement tracking systems. The developed Knowledge Graph (KG) approach was based on this reference model that integrates manufacturing objectives (e.g. improve efficiency, productivity reliability, and quality), activity (manufacturing processes and workflows), resource (e.g. machines, tools, consumables, controllers, and workers), environment (e.g. factory layout, safety, working conditions), factory movement (e.g. location, direction of movement, velocity, time), data (e.g. data types, format, secure), infrastructure (e.g. sensors, data collection, storage, transfer) and decision support system (e.g. feedback systems, descriptive, prescriptive, predictive). The reference model facilitates mapping the research abstracts into these entities and transforming them into the KG. Figure 2 describes the structure and possible influences on these entities. The figure also includes the subcategories within these entities.

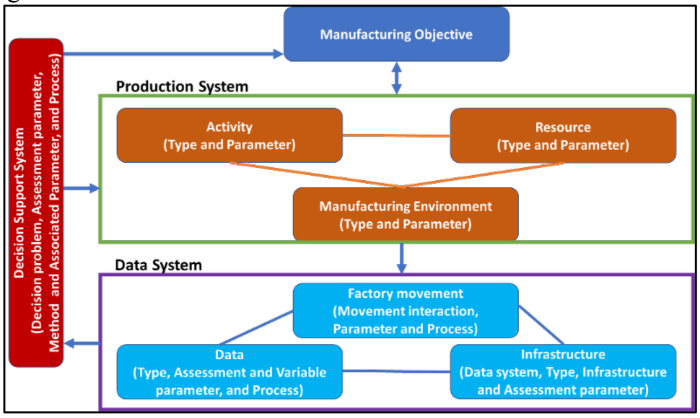

Fig.2. Knowledge representation reference model

5 Knowledge Graph Generation Approach

The Knowledge Graph (KG) aims to create a semantic network by interlinking concepts, entities, and relationships related to factory movement system implementation. Four steps are structured to convert a research abstract into a KG:

Step 1: Keywords/phrases extraction from the research abstract.
Step 2: Mapping keywords to established entities in the reference model.
Step 3: Mapping keyword relationships based on object properties.
Step 4: Overall coherence check in the KG development.

Since accurate interpretation and consistent development are vital, these steps were conducted manually. However, once a significant labelled dataset is available, it will facilitate the development of an automated approach. The significance of the labelled dataset will be established by assessing the performance of the KG in answering queries correctly and comprehensively. Validating and updating labelled datasets regularly will improve the robust development of the KG. The following paragraphs detail each step using the below mentioned research abstract from Addison and Macleod [13]:

"We describe a new tracking algorithm for the direction of arrival estimation problem where both the locations of the sensors in the array and the directions of arrival are nonstationary. The approach taken is Bayesian. The algorithm assumes that the filtering distribution is approximately Gaussian and maintains the mean and covariance of this approximation by fitting a quadratic surface to the log posterior around the location where the log posterior is maximized. In the case where the sensor locations are stationary, the algorithm is shown to have similar performance to particle filter-based algorithms but at a reduced computational cost. In the case where the sensor locations are non-stationary particle filtering is unsuccessful and the new algorithm performs significantly better than currently existing algorithms."

Step 1: Keywords/phrases extraction: A keyword, or phrase, is a single word or a set of words that provide meaningful scientific content. It could be a subject or object in the sentence description. The predicate (i.e., properties/characterizing the keyword/phrases) was ignored at this step (e.g. 'reduced' was not included in 'reduced computational cost'). Key phrases were redefined to provide compact representation (e.g., 'locations of the sensors' into 'sensor location'). Some key phrases were maintained as they were if that's how the scientific community termed it (e.g. direction of arrival (DoA)). All extracted keywords/phrases were represented as singular (Table 1). Acronyms were expanded, and repeated keywords/phrases and synonyms were ignored. Also, generic words (such as performance) were not included.

Table 1. Extracted Keywords/phrases from the abstract

Extracted Keywords/phrases		
Tracking algorithm	Filtering distribution	Covariance
Bayesian approach	Gaussian	Stationary
Computational cost	Mean	Particle filter-based algorithm
Direction of arrival location	Direction of arrival estimation problem	Log posterior maximized location
Nonstationary	Fitting Quadratic surface	Sensor location in the array

Step 2: Mapping keywords to established entities in the reference model. In this step, the extracted keywords were mapped to the entities represented in the reference model (Table 2). The subcategories within the reference entities were considered and mapped.

Table 2. Mapping keywords/phrases to Reference entities

Keywords/phrases	Mapped Reference model entity	Keywords/phrases	Mapped Reference model entity
Tracking algorithm	Decision method	Filtering distribution	Decision process
Direction of arrival estimation problem	Decision problem	Nonstationary	Infrastructure parameter
Sensor location in the array	Infrastructure parameter	Direction of arrival location	Movement parameter
Computational cost	Decision Assessment parameter	Gaussian	Data variable parameter
Mean	Data type	Covariance	Data type
Stationary	Infrastructure parameter	Fitting Quadratic surface	Decision process
Log posterior maximized location	Decision associated parameter	Particle filter based algorithm	Decision method
Bayesian approach	Decision method		

Step 3: Mapping keyword relationships based on object properties. The relationships used in this abstract are *'apply to', 'assume', 'characterized by, 'has type', 'is better than', 'maintain by', 'reduce', 'same performance', 'near by',* and *'utilize'*. Identifying the subject, predicate, and object in a sentence help to find appropriate relationships between keywords. For example, the relationships identified from the first sentence in the abstract are:

"We describe a new tracking algorithm for the direction of arrival estimation problem where both the locations of the sensors in the array and the directions of arrival are nonstationary."

Tracking algorithm → (apply to) → Direction of arrival estimation problem → (characterize by) → Sensor location in the array and direction of arrival location → (has type) → Non-stationary.

The above example utilized the predefined object properties for consistent KG development. New object properties could be added if the existing properties do not represent that object relationship. Class expressions describe relationships between key phrases using the identified object properties. The class expression syntaxes utilized in Protégé are: *some, value, only, min, max, exactly, and, or, and not*. Complex class expressions could describe the intricate relationships between key phrases and object properties. The representation of the complex abstract statement by the reference model entities is shown below:

"In the case where the sensor locations are stationary, the algorithm is shown to have similar performance to particle filter-based algorithms but at a reduced computational cost."

In KG definitions, the Tracking algorithm included the following Sub-Class relationship entities:

(same_performance some Particle_Filter_based_Algorithm) and (has_type only Stationary).

(is_better_than some Particle_Filter_based_Algorithm) and (reduce some Computational_Cost).

The above example effectively demonstrates the building of complex class expressions representing key phrases and object relationships. In addition, class expressions can be nested to any required depths to build up detailed descriptions.

Step 4: Overall coherence check in the KG development. Although sentence-based analysis establishes explicit relationships between key phrases, implicit relationships (not directly represented in the abstract) should also be identified. The following implicit object relationships have been added for completion in the example abstract.

Particle Filter-based algorithm → (apply to) → Direction of Arrival Estimation.

Sensor location in the array → (has type) → Stationary.

The coherence check will ensure all keywords/phrases are related to appropriate object relationships. No keyword should be unconnected. Finally, the identified keywords and relationships are visualized in OntoGraf within Protégé software, which logically facilitates checking the KG structure. Figure 3 illustrates the KG developed from the example abstract. The first three rows define the direction of arrival estimation problem, the fourth row describes the methods used and assessment parameters, and the rest describes the proposed method.

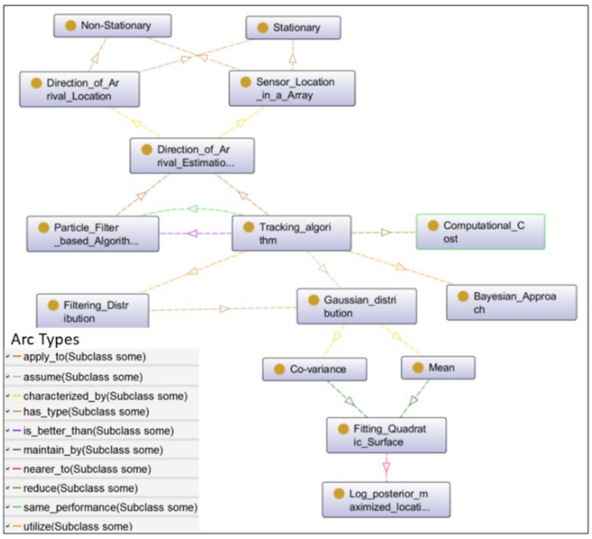

Fig. 3. KG representation of the example abstract using OntoGraf[1]

Following the same KG generation approach, 30 further research abstracts related to the factory movement tracking system were analyzed, mapped and synthesized into a single Protégé ontological file. This OWL protégé file can be accessed from this link[2]. This file could be used to visualize the complete KG in WebVOWL[3].

6 Knowledge Graph Assessment and Validation

This Knowledge Graph (KG) can help answer questions about implementing a factory movement tracking system, whether general or specific to technical details. Several query tools are available in Protégé software, such as DLQuery and SPARQL, that can be used to extract information from the KG. Some examples of queries and their results are presented in this section.

Specific Queries:

- What are the applications of the Affine Iterative Closest Point Method?
- What parameters does Affine Iterative Closest Point Method utilize?
- Do the Affine Iterative Closest Point Method better than the other method?
- Which article mentioned Affine Iterative Closest Point Method?
- How was the Affine Iterative Closest Point Method assessed?

The SPARQL Query for answering all these questions is:

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX ss: <http://www.semanticweb.org/40013116/ontologies/2022/5/untitled-ontology-11#>

SELECT ?object

---

[1] https://protegewiki.stanford.edu/wiki/OntoGraf
[2] http://shorturl.at/pyMN3
[3] http://vowl.visualdataweb.org/webvowl.html

WHERE {ss:Affine_Iterative_Closest_Point_Method rdfs:subClassOf ?object };

Figure 4 depicts the results obtained for the above query answering all the above-mentioned specific queries. The search query results are correct and will be useful in exploring methods and tools for implementing the movement tracking system.

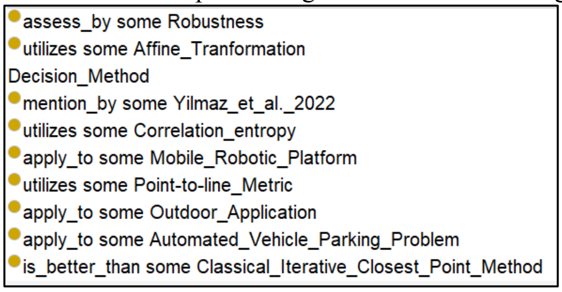

Fig. 4. SPARQL query results

A researcher tasked with evaluating the resulting KG for factory movement tracking systems made the following remarks:

- The KG was beneficial for understanding the state-of-the-art and quickly revealed the domain by viewing key entities such as decision problems and manufacturing objectives.
- The classification structures were appropriate for the entities. However, the researcher observed that overlaps between different classes are inevitable.
- The notable feature mentioned is that the KG provides a wide range of knowledge-based relationships between entities that are not necessarily hierarchic.
- The researcher noted that exploring object relationships supports identifying research gaps in the manufacturing factory movement tracking domain.

7 Discussion and Conclusion

This research presents a KG approach for the state-of-the-art implementation of an industrial factory movement tracking system. A knowledge representation reference model was proposed to support generating the KG. The resulting KG, which was created based on the analysis of 30 research abstracts, offers several advantages, including fast search and question answering, improved understanding of the current state-of-the-art in the movement tracking domain, identification of research gaps, and the ability to explore a wide range of knowledge-based relationships. Although the manual approach accurately applies the KG generation steps, the process is time-consuming to generate, validate, and update the structured KG. Also, analyzing only the research abstracts has limitations in covering overall knowledge comprehensively.

Further, the KG developer requires an understanding of the movement tracking system for correctly mapping keywords/phrases to the proposed reference entities. Significant progress has been made in automated knowledge discovery, such as Amazon AutoKnow [14], which built KGs for product descriptions. However, compared to known structured data description, creating KGs for a specific research domain is a challenging process for the following reasons: the sparsity/new terminologies used across varying research articles; a multitude of interpretations of reported findings; context understanding, complex and evolving nature of knowledge relationships; broader scope of synonym check, and extracting implicit relationships between entities. Therefore, the authors believe that manual and automated approaches should co-evolve, where annotated texts by the manual process could be helpful to train and develop an automated method for generating research-specific KGs. The ongoing research of this work is to develop the KG comprehensively by analyzing more research abstracts and evaluating the efficiency of automated computing approaches in developing this knowledge graph.

Acknowledgements

This work was supported by the Engineering and Physical Sciences Research Council, UK [grant number EP/V051113/1 - Productivity and Sustainability Management in the Responsive Factory].

References

1. Ing, T. S. et al. An overview of the rising challenges in implementing industry 4.0. International Journal of Supply Chain Management, 8(6), 1181-1188 (2019).
2. Moktadir, M. A. et al. Assessing challenges for implementing Industry 4.0: Implications for process safety and environmental protection. Process safety and environmental protection, 117, 730-741 (2018).
3. Müller, J. M. et al. What drives the implementation of Industry 4.0? The role of opportunities and challenges in the context of sustainability. Sustainability, 10(1), 247 (2018).
4. Veile, J.W. et al. Lessons learned from Industry 4.0 implementation in the German manufacturing industry, Journal of Manufacturing Technology Management, Vol. 31 (2020).
5. Nakagawa, E. Y. et al. Industry 4.0 reference architectures: State of the art and future trends. Computers & Industrial Engineering, 156, 107241 (2021).
6. Mirani, A. A. et al. Key Challenges and Emerging Technologies in Industrial IoT Architectures: A Review. Sensors, 22(15), 5836 (2022).
7. Anumbe, N. et al. A Primer on the Factories of the Future. Sensors, 22(15), 5834 (2022).
8. Bloom, G. et al. Design patterns for the industrial Internet of Things. 14th IEEE International Workshop on Factory Communication Systems pp. 1-10 (2018, June).
9. Washizaki, H., et al. Landscape of IoT patterns. IEEE/ACM 1st International Workshop on Software Engineering Research & Practices for the Internet of Things pp. 57-60 (2019).
10. Compton, M., et al. The SSN ontology of the W3C semantic sensor network incubator group. Journal of Web Semantics, 17, 25-32 (2012).
11. IEC 62264-1:2003, Enterprise-control system integration -Part 1: Models and terminology.
12. National Institute of Standards and Technology, NIST, The Process Specification Language, March (2005).
13. Addison, W. D., Macleod, M. D. Non-stationary Bayesian direction of arrival estimation with drifting sensor locations. 16th European Signal Processing Conference pp.1-5 (2008).
14. Dong, X. L. Autoknow: Self-driving knowledge collection for products of thousands of types. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining pp. 2724-2734 (2020, August).