5th Conference on Production Systems and Logistics

Building A Knowledge Graph From Deviation Documentation For Problem-Solving On The Shop Floor

Yuxi Wang [1], Beatriz Bretones Cassoli[1] , Joachim Metternich[1]
[1]Institute of Production Management, Technology and Machine Tools, Darmstadt, Germany

Abstract

The description of deviations on the shop floor includes information about the deviation itself, possible causes and countermeasures. This information about current and already processed deviations and problems is a valuable source for future activities in the context of problem-solving and deviation management. However, extracting information from unstructured textual data is challenging. Furthermore, the relationships among the heterogeneous data are hard to represent. This paper proposes a framework to extract the knowledge contained in the deviation documentation and store it in a knowledge graph as triples. The proposed knowledge graph can then be used for the decision support system in production and will support more application scenarios in shop floor management in the future.

Keywords

Shop Floor Management; Deviation Management; Knowledge Graph; Decision Support System.

1. Introduction

The documentation of deviations serves as a valuable source of knowledge on the shop floor, encompassing various aspects such as problem-solving, production planning, and continuous improvement process [1]. It connects numerous entities involved in the production process, including people, processes, organizations, machines, and products. This documentation is especially useful for managers and workers on the shop floor as it aids in identifying significant deviations, exploring measures of similar deviations, and finding useful information [2]. However, processing the deviation documentation faces several challenges. One of the primary challenges is dealing with heterogeneous data, which comprises structured and unstructured information [1]. Especially the textual data, which contains crucial entities of deviations, is difficult to standardize due to the specificities of the production environment [3]. Moreover, the textual data often contains errors, making it challenging to accurately process and extract valuable insights from it [4]. Therefore, extracting representations of the relationships between deviations, measures, causes, and other relevant information from such documents containing unstructured data is challenging. Without this relational structure, the person accessing the information would have to search the records item by item using keywords, which would make it difficult to reuse the data. Over time, this knowledge of experience loses its value [5].

As a tool for organizing and integrating different knowledge and information, the knowledge graph (KG) has been a focus of research in the computer field since it was introduced by Google in 2012. It is capable of knowledge tracing and reasoning, providing significant assistance in decision support across various domains [6]. In the industrial sector, with the continuous advancement of digitization processes, industrial data has exhibited heterogeneous characteristics. As more and more application scenarios can be supported by data, the application of KG in the industrial sector is also gradually increasing. A KG offers a structured

DOI: https://doi.org/10.15488/15244
ISSN: 2701-6277
publish-Ing.

and interconnected representation of information, enabling better decision-making, knowledge reuse, and the development of advanced recommender systems to drive continuous improvement and efficiency in the production process [7]. It can support SFM from providing the possible measures, clustering the recurring deviations, visualising the problem-cause-solution loop and more application scenarios. Therefore, the goal of this paper is to design a framework for building a KG from deviation documentation for problem-solving on the shop floor. Based on this, the major research contributions of this work can be summarized as follows:

- A stepwise process is proposed and designed for constructing a KG from shop floor deviation documentation.
- Named entity recognition (NER) and rule-based methods are used to extract entities and relations from the unstructured data in the deviation documentation.
- Information from structured and unstructured data is fused into one KG for further problem-solving on the shop floor.

The remainder of this paper is structured as follows. Section 2 introduces the background and related work of this study. Section 3 presents the framework to build a KG including each step. Section 4 shows the detailed steps of KG construction with data from deviation documentation on the shop floor. Finally, Section 5 concludes this study and suggests directions for future research.

2. Related Work

2.1 Shop floor management

Shop floor management (SFM) is widely implemented in manufacturing to control and improve production processes continuously [8]. The basic elements of SFM included identifying and handling deviations, systematic problem-solving process for the deviation with unknown root causes, information and knowledge exchange in the regular shop floor meetings, and continuous improvement process [9]. For deviations that occur in production, if the cause of the deviation is unknown, the root cause should be analyzed through a problem-solving process so that it can be solved sustainably in a long-term perspective and does not reoccur [10]. Furthermore, while the problem-solving process can bring new standards to the processes on the shop floor, these criteria form the basis for new deviation measurement [11]. Deviation management and problem-solving process are two knowledge-intensive processes in SFM that require much experience to aid decision support [12]. Over the last years, many articles have emphasized the importance of data analysis for deviation management and problem-solving processes, and there has been some research on how to extract useful information from shop-floor management data for decision support [13,14]. However, fewer articles have mentioned how to correlate the different knowledge involved in this domain specific process with each other to provide a good database for knowledge retrieval and knowledge inference.

2.2 Knowledge graph

A KG is a data representation modality in which entities (depicted as nodes in the graph) are connected to other entities through edges. Edges describe the relationships which connect and relate these entities to each other [15]. A KG adds a layer of metadata to the data (also called semantic context), defining rules for its structure and interpretation [6]. They can, therefore, represent complex relationships in a domain in both machine friendly and human-readable forms to support reasoning and knowledge discovery [6].

Modelling data as a KG is particularly useful when the relationships within the data help to understand and solve the problem at hand [16]. KG can thus be used in a variety of applications, such as to generate reports from the data, provide a service to an end user in a retrieval system (e.g., in question answering and recommender systems), and also as part of machine learning pipelines [17]. The main task of KG construction is to reduce the granularity of data and avoid data redundancy while aggregating a large amount of knowledge, so that rapid response and inference of knowledge can be realized. KG construction methods are mainly categorized into top-down, bottom-up and hybrid methods [18,19]. Among them, generalized KG are mainly constructed using a bottom-up approach, i.e., entities and relationships are extracted from the data, and then ontologies are constructed. Domain KG, on the other hand, are more complex and also use a top-down or hybrid approach to narrow down the data by defining entities in advance. [19] The construction of a KG will generally be divided into the following processes: data acquisition, information extraction, knowledge fusion, quality control and graph construction [18,19,20].

2.3 Ontology

Ontology is a formal, explicit specification of a shared conceptualization, allowing a non-ambiguous semantic explanation of domain knowledge, enabling a better representation of the knowledge [21]. Ontologies define important structures for KG. Literature shows different ontologies related to deviation management and problem-solving. Ebrahimioour et al. propose an upper ontology where three concepts related to the failure modes and effects analysis information (i.e., deviation, cause and consequence) are modelled as an event and activities [22]. A deviation is modelled as an event, the beginning of a consequence. A consequence is an activity. A cause is an activity that causes a deviation. Dittmann et al. propose an ontology that is specialized into the entities such as component, function, failure mode, control method, risk priority number, and containment action [23]. They also propose a set of relationships among the entities. After reviewing the mentioned ontologies, the work from Ebrahimioour et al. were taken as reference. Section 4.2 explains the selected concepts that were adopted and the new concepts that are proposed, which are part of the contribution of this work.

3. Framework to build a KG from deviation documentation

Figure 1 presents the process of building a KG from the deviation documentation from the shop floor. The framework's design is based on the top-down KG construction process, as top-down KG construction is more suitable for domain knowledge [20]. This framework comprises five main tasks: data acquisition, ontology construction, knowledge extraction, knowledge fusion and KG construction.

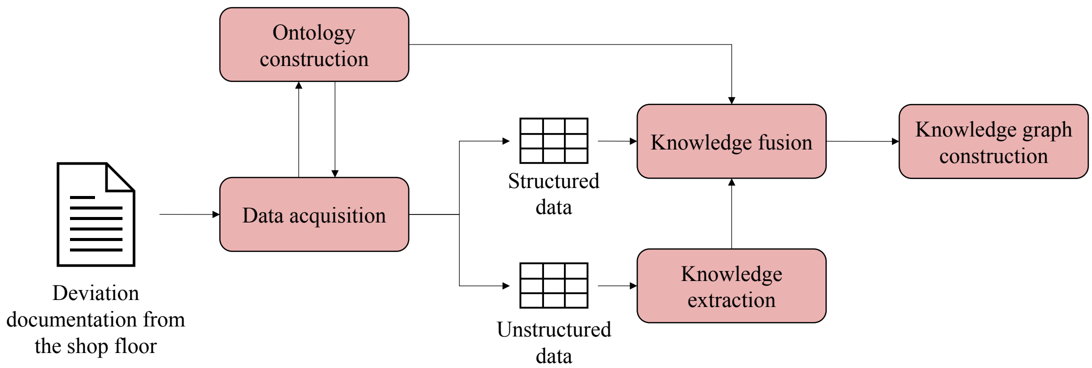

Figure 1 The process to build the knowledge graph

Data acquisition describes the data collection required for KG, which may involve structured or unstructured data. Structured data refers to data with strict structure [24], such as the deviation type, product type or date. Unstructured data include the deviation description, figures or videos containing detailed information about the deviation. As the KG for deviation management is domain-specific, the data sources should be identified and analyzed during the data acquisition.

Ontology construction is used to build a semantic explanation of the domain knowledge, which provides good instruction for knowledge extraction and a structure for the KG construction [7]. In this paper's proposed framework, the KG construction process can be started by either data acquisition or ontology construction, and there may be reciprocity in both processes. Since both deviation management and problem-solving process are essential components of SFM, their theoretical elements should be able to help accomplish ontology construction directly. There is also much research in the literature on how to build ontologies for the problem-solving process. However, in practice, a large amount of data related to deviation and problem-solving is knowledge-intensive, making ontology construction much more difficult [25]. Whether this data already exists in the system, in what system it is stored, and whether it is stored in a structured or unstructured form, the answers to these questions can vary greatly depending on the actual situation. It is, therefore, necessary to continuously adapt the ontology as it is being constructed to the actual situation and end use of the KG.

Knowledge extraction in the construction process of a KG mainly targets semi-structured and unstructured data. Its purpose is to discover valuable information within unstructured data and extract it. Knowledge extraction typically involves three tasks: entity extraction, relation extraction, and attribute extraction [20]. Since this process often deals with textual data, named entity recognition is a primary natural language processing method used for knowledge extraction [26].

Knowledge fusion mainly focuses on the integration of different data. In addition to the structured and unstructured data mentioned earlier, this data can also include third-party data from external sources. Another key task of knowledge fusion is to eliminate ambiguity in knowledge, such as merging entities and relationships that represent the same content.

KG construction involves integrating all the data based on a predefined ontology structure into a graph-based database, and it can be visualized in a graphical format.

4. Building a KG from deviation documentation

In order to better illustrate how the framework mentioned in Chapter 3 can be utilized to build a KG, we validate the KG construction process by using deviation documents from the shop floor of an industrial company in Germany as a database.

4.1 Data acquisition

In data acquisition, we perform both data cleaning and data understanding. The primary purpose of data cleaning is to remove meaningless or unclear content in the data. On this basis, the data can be better understood, which enables better ontology construction in the next step. In this process, it is also necessary to distinguish between structured and unstructured data and to analyse what additional information and knowledge unstructured data can bring to structured data.

4.2 Ontology construction

Based on the data acquisition and analysis in Section 4.1, we construct an ontology structure, as shown in Figure 2. This ontology structure is centred on the occurrence of deviations. It contains four types of core content related to them, three of which can be obtained directly from the structured data, which are the information related to the classification of deviations, the information on the location where the deviation occurred, and the information on the product where the deviation was found. This covers seven different entities and four different relationships. The last piece of content is the information related to the deviation event, which includes the symptoms at the time, the causes of the deviation, and the contra measures of the deviation after it has occurred. The employees record these data in the form of unstructured text in the deviation documents and need to be extracted into the KG through knowledge extraction. This part includes three entities and four relationships. In addition, the deviation documentation contains other information related to the deviation, such as the costs incurred by the deviation. Since this information is associated with a single deviation, it is stored as attribute in the deviation node. The details will be discussed in Section 4.4.

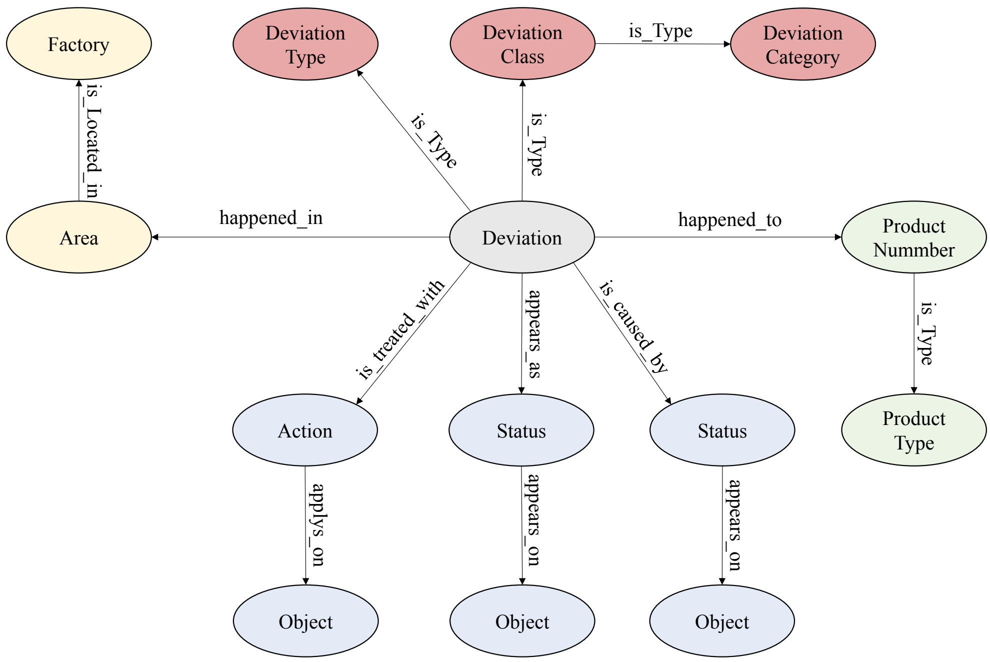

Figure 2 KG structure of the deviation documentation

4.3 Knowledge extraction

As mentioned in Section 3, the structured data already has a specific data model that can directly convert to entity, relation and attribute in the KG. Therefore, knowledge extraction focuses on the unstructured data in the deviation documentation, i.e., the descriptions of symptoms, causes, and measures. Based on the analysis of the writing characteristics, three types of node entities are set up, namely *Object*, *Status* and *Action*, where *Status* and *Object* can describe symptoms and causes. The measure needs to be described through *Action* and *Object*. Four relationship entities are set as *is_treated_with*, *appears_as*, *is_caused_by* and *appears_on*. The process and result of extracting node entities and relationship entities are shown in Sections 4.1.1 and 4.1.2.

4.3.1 Entity extraction

The task of entity extraction is to recognize all the deviation related objects, status and action in the unstructured data, so that the information can be stored in a structural way. As mentioned in Section 3, NER is used as a tool to complete the task. In the dataset given the total number of text data records involving deviation symptoms, causes, and measures is 5518. There are 2501 deviation symptoms, 1497 causes, and 1520 measures. The entities are labelled using the open-source annotator "NER Annotator for SpaCy" and stored in json file using BILOU structure as shown in Figure 3 [11].

English	Plug glued
German	["Stecker verklebt\r",{"entities":[[0,7,"OBJECT"],[8,16,"STATUS"]]}]

Figure 3 Annotation result in BILOU structure

In this paper, a NER model was created using the spaCy v3.6 named entity recognition system. The model architecture consists of a two-layer pipeline: a context embedding layer and a transformation-based chunking model (cite). The first layer uses a pre-trained language model to encode tokens into continuous vectors based on context. The second model predicts text structure by mapping it to a set of state transitions. It uses the output of the previous step (contextual word embeddings) to incrementally construct states from the input sequence and assigns entity labels to them using a multilayer neural network. We trained and compared two spacy-based entity recognition pipelines using German-BERT [27] and XLM-RoBERTa [28] as contextual embedding layers. The ratio of training set, test set, and validation set is 3:1:1. We use precision, recall, and F1-Score as evaluation metrics. The calculation formula is as follows:

$$Precision = \frac{\sum_{k=1}^{n} \text{The number of correctly predicted labels in sentence}_k}{\sum_{k=1}^{n} \text{The number of predicted labels in sentence}_k}$$

$$Recall = \frac{\sum_{k=1}^{n} \text{The number of correctly predicted labels in sentence}_k}{\sum_{k=1}^{n} \text{The number of labels that should be included in sentence}_k}$$

$$F_1 - Score = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{Precision + Recall}$$

The training results of each model on the test set are shown in Table 1. Overall, XLM-RoBERTa + spaCy has the better prediction performance.

Table 1. Performances of different models on entity extraction related to symptoms

	German-BERT + spaCy			**XLM-RoBERTa + spaCy**		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Symptoms	87.76%	83.50%	85.57%	**88.66%**	**86%**	**87.31%**
Causes	79.17%	69.72%	74.15%	**87.76%**	**83.50%**	**85.57%**
Measures	83.51%	76.42%	79.80%	**88.21%**	**84.73%**	**86.43%**

4.3.2 Relation extraction

Rule-based relation extraction methods are suitable for text-poor linguistic environments [29]. However, unlike many other relation extraction tasks, the text in deviation documents are mostly short texts and the syntax of sentences is often incomplete, with no words between entities that can be used to construct regular relations, making it unsuitable for mining relations using a relational triple extraction. By analysing the linguistic patterns in the data, this paper summarizes three typical relationships. For symptoms, causes and measures the patterns are similar:

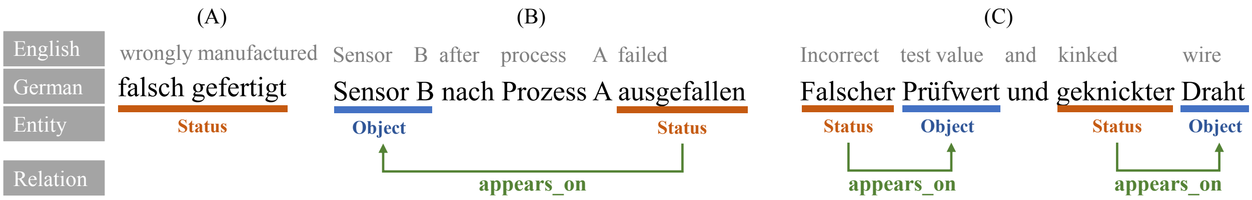

Figure 4 Patterns recognition for rule-based relation extraction (with symptoms as examples)

(A) *Status*/*Action(s)* without *Object*: in these cases, the deviation can directly link to the *Status* or *Action(s)* with the relationship *is_treated_with*/*appears_as*/*is_caused_by*.

(B) One *Status*/*Action* with more *Objects*: in these cases, the relationship *appear*_on between *Status* and *Object* is one-to-many.

(C) More *Status*/*Action(s)* with more *Objects*: there are many different possibilities for this scenario. In order to simplify the process of relation extraction, this paper analyzes the situation that comes out of most texts and decides to assign relations on a one-to-one basis according to position. Here, the relations between *Objects* and *Status* that are in the same order are one-to-one.

4.4 Knowledge fusion

In the knowledge fusion process, the main task is to connect structured and unstructured data. Here, we first process the structured data and establish relationships between them. Tables 2-5demonstrate how the structured data is specifically stored in entity nodes related to deviation, location and product.

Table 2 Entities about deviation

Type	Label	Property	Example	Number of Entities
Node entity	Deviation	Name	1713	4052
		Number of Scrap	1	
		Unit cost	317,88 €	
		Total cost	317,88 €	
		Date	26.03.2020	
	Deviation Type	Name	Masseschluss (Ground fault)	44
		Code	443	
	Deviation Class	Name	Arbeitsbeschädigt (Work-damaged)	34
	Deviation Category	Name	Produktion (Production)	10
Relation entity	is_Type (from Deviation to Deviation Type)	Name	is_Type	4035
	is_Type (from Deviation to Deviation Class)	Name	is_Type	4031
	is_Type (from Deviation Class to Deviation Catogory)	Name	is_Type	36

Table 3 Entities about location

Type	Label	Property	Example	Number of Entities
Node entity	Area	Name	Übertrager (Transformer)	13
	Factory	Name	1002	5
Relation entity	is_Located_in (from Area to Factory)	Name	is_Located_in	26

Table 4 Entities about product

Type	Label	Property	Example	Number of Entities
Node entity	Product No.	Name	847152694	1176
	Product Type	Name	TRIEBWERK 122 (ENGINE 122)	572
		Serial Number	4-85471-885	
Relation entity	is_Type (from Product No. to Product Type)	Name	is_Type	1024

Table 5 Entities between deviation, location and product

Type	Label	Property	Example	Number of Entities
Relation entity	happened_in (from Deviation to Area)	Name	happened_in	4038
	happened_to (from Deviation to Product No.)	Name	happened_to	1259

4.5 KG construction

Using the py2neo toolkit, data is stored in the Neo4j graph database, and the visualization results in the following graph as shown in Figure 5. After importing the deviation-related, location-related and product-related data from the deviation documentation into KG, 9277 nodes and 22297 relations were created.

Figure 5 The KG from deviation documentation in neo4j (deviation at the middle)

Figure 5 shows an ontology-similar KG with one deviation in the center. With the KG, the deviation related information can be stored in with a graph-based database. In addition to store the information in KG in a structured way, the KG from deviation documentation can achieve more convenient query and statistic functions. For example, if workers on the shop floor have the deviation relevant to “Electronic” and “defect”, they can use the KG to find possible measures in the past. The query statement can be “MATCH p=(ac:Action)<-[]-()-[]->(s:Status{statusId:"defekt"})-[]->(o:Object{objectId:"Elektronik"}) RETURN p”. As shown in Figure 6, the *Action* “exchanged” pointing on the *Object* “Electronic” is found.

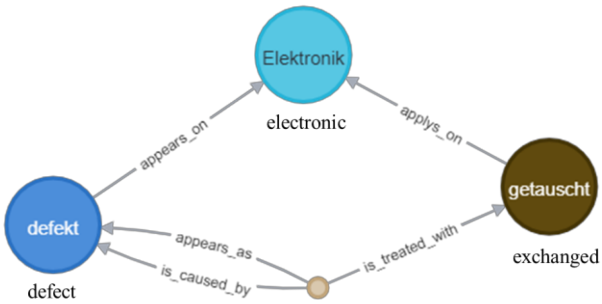

Figure 6 The KG from deviation documentation in neo4j (searching for measures)

Furthermore, the KG can help cluster and analyze the deviations for one product type. With the query “MATCH p=()<-[]-()<-[r:appears_as]-()-[]->()-[]->(pt:ProductType{producttypeId:"ROTOR 13558"}) RETURN p”, the KG can find out all the *Status* and their related *Object* with the relation *appears_as* in between as connection. As shown in Figure 7, all the deviations that have happened to a product type are automatically clustered.

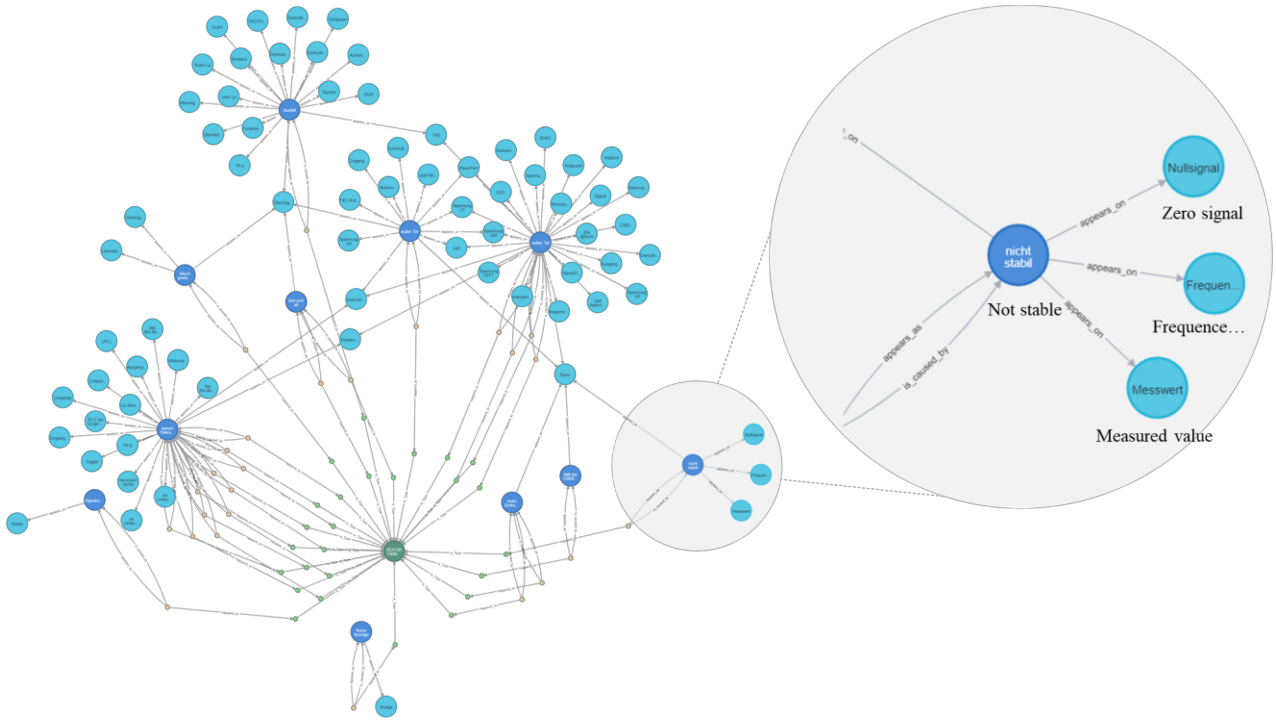

Figure 7 The KG from deviation documentation in neo4j (categorizing the deviations for a product type)

The KG of deviation management can serve as a data foundation for many new application scenarios in the future. Firstly, it enables visualization of the knowledge involved in the deviation management. With the queries, the information and data for certain usage can be found out based on the keywords. It is no longer necessary to browse through complex deviation archives; instead, the past deviations can be categorized by extracting the graph, which leads to a better understanding. The connections between different pieces of information also enable knowledge retrieval. By simply issuing straightforward command statements, the possible causes and solutions to deviations can be searched, which leads to the reduction of response time. Moreover, the KG can be utilized with artificial intelligence algorithms for predictions, such as forecasting the potential impact of newly occurring deviations, providing decision support for more activities in the SFM.

5. Conclusion and Outlook

This paper describes how to build a KG in SFM from the information in deviation documents. To this end, in this article framework is designed for building a top-down KG based on the data characteristics and target application scenarios in SFM. In the process of knowledge extraction, this paper uses the NER approach to extract the entities out of the unstructured textual data with spacy-based entity recognition pipelines using German-BERT and XLM-RoBERTa as contextual embedding layers. The results showed that compared to German-BERT, the XLM-RoBERTa as embedding layers had a better performance. For extracting the relations, this paper designed a rule-based method subject to the language specifications. After completing the knowledge fusion for structured data and unstructured data, this paper presented the KG in Neo4j and illustrated the futural usages of the KG for the problem-solving process on the shop floor.

Due to the specificity of shop floor deviation management and problem-solving data, some improvement space has been identified in this paper during the study of KG construction techniques, which can be continued through future research. For example, the process of detecting deviations, handling deviations and solving deviations contains not only the state of the product and the production line (time point), but also the attempts made by the workers to understand the deviations and to fix them (events with time periods). If these events and states can also be represented in the KG as entities and connected with relations, it can greatly help the decision support process for future problem solving. Furthermore, the large language models such as the Large Language Model Meta AI (LLaMA) and the Generative Pre-trained Transformer (GPT) can also be tested for the NLP tasks in KG construction processes. It should be evaluated whether the large language models can have a better performance on the domain specific data such as the data from shop floor.

The use of KG in shop floor-related activities is not yet widespread, but many possibilities exist. One example is evaluating the deviations through the complete information of the deviations in the KG, determining which deviations need to be handled with systematic problem-solving, and determining their prioritization. A knowledge backtracking of KG can also be designed to provide solutions for newly recognized deviations in production. Graph-based algorithms can be used to predict the possible impact of newly occurring deviations to trigger the proactive measures to prevent the deviations. These future application scenarios for KG are well worth continuing to investigate.

Acknowledgement

This research is funded by the European Institute of Innovation & Technology (EIT) Manufacturing.

References

[1] Wang, Y., Longard, L., Hertle, C., Metternich, J., 2023. Beyond Pareto Analysis: A Decision Support Model for the Prioritization of Deviations with Natural Language Processing. Hannover: publish-Ing, 11 pp.

[2] Müller, M., Alexandi, E., Metternich, J., 2021. Digital Shop Floor Management Enhanced by Natural Language Processing. Procedia CIRP 96, 21–26.

[3] Müller, M., Frick, N., Lee, J.-U., Metternich, J., Gurevych, I., 2020. Chats als Datengrundlage für KI-Anwendungen in der Produktion. Zeitschrift für Wirtschaftlichen Fabrikbetrieb : ZWF 115 (7-8), 520–523.

[4] Müller, M., Metternich, J., 2021. Production Specific Language Characteristics to Improve NLP Applications on the Shop Floor. Procedia CIRP 104, 1890–1895.

[5] Müller, M., Terziev, G., Metternich, J., Landmann, N., 2020. Knowledge Management on the Shop Floor Through Recommender Engines. Procedia Manufacturing 52, 344–349.

[6] Barrasa, J., Hodler, A., Webber, J., 2021. Knowledge Graphs. O'Reilly Media, Incorporated.

[7] He, L., Jiang, P., 2019. Manufacturing Knowledge Graph: A Connectivism to Answer Production Problems Query With Knowledge Reuse. IEEE Access 7, 101231–101244.

[8] Pötters, P., Schindler, P., Leyendecker, B., 2018. Status quo Shopfloor Management. Zeitschrift für wirtschaftlichen Fabrikbetrieb 113 (7-8), 522–525.

[9] Suzaki, K., 1993. The new shop floor management: Empowering people for continuous improvement. The Free Press, New York, 462 pp.

[10] Peters, R., 2009. Shopfloor Management: Führen am Ort der Wertschöpfung. LOG_X, Stuttgart, 141 pp.

[11] Hertle, C., Tisch, M., Metternich, J., Abele, E., 2017. Das Darmstädter Shopfloor Management-Modell. Zeitschrift für Wirtschaftlichen Fabrikbetrieb : ZWF, Carl Hanser Verlag 112 (3), 118–120.

[12] Choo, A.S., Nag, R., Xia, Y., 2015. The role of executive problem solving in knowledge accumulation and manufacturing improvements. Journal of Operations Management 36 (1), 63–74.

[13] Longard, L., Schiborr, L., Metternich, J., 2022. Potentials and obstacles of the use of data mining in problem-solving processes. Procedia CIRP 107, 252–257.

[14] Lin, H.K., Harding, J.A., 2007. A manufacturing system engineering ontology model on the semantic web for inter-enterprise collaboration. Computers in Industry 58 (5), 428–437.

[15] Robinson, I., Webber, J., Eifrem, E., 2015. Graph databases: new opportunities for connected data. " O'Reilly Media, Inc.".

[16] Gosnell, D., Broecheler, M., 2020. The Practitioner's Guide to Graph Data: Applying Graph Thinking and Graph Technologies to Solve Complex Problems. " O'Reilly Media, Inc.".

[17] Bretones Cassoli, B., Jourdan, N., Metternich, J., 2022. Knowledge Graphs for Data And Knowledge Management in Cyber-Physical Production Systems, in: Proceedings of the Conference on Production Systems and Logistics: CPSL 2022. Hannover: publish-Ing, pp. 445–454.

[18] Danling Yuan, Keping Zhou and Chun Yang. Architecture and Application of Traffic Safety Management Knowledge Graph Based on Neo4j.

[19] Chicaiza, J., Valdiviezo-Diaz, P., 2021. A Comprehensive Survey of Knowledge Graph-Based Recommender Systems: Technologies, Development, and Contributions. Information 12 (6), 232.

[20] Tamašauskaitė, G., Groth, P., 2023. Defining a Knowledge Graph Development Process Through a Systematic Review. ACM Trans. Softw. Eng. Methodol. 32 (1), 1–40.

[21] Giovannini, A., Aubry, A., Panetto, H., Dassisti, M., El Haouzi, H., 2012. Ontology-based system for supporting manufacturing sustainability. Annual Reviews in Control 36 (2), 309–317.

[22] Ebrahimipour, V., Rezaie, K., Shokravi, S., 2010. An ontology approach to support FMEA studies. Expert Systems with Applications 37 (1), 671–677.

[23] Dittmann, Rademacher, Zelewski. Performing FMEA Using Ontologies.

[24] Tianxing Wu, Guilin Qi, Cheng Li and Meng Wang. A Survey of Techniques for Constructing Chinese Knowledge Graphs and Their Applications.

[25] Ansari, F., Khobreh, M., Seidenberg, U., Sihn, W., 2018. A problem-solving ontology for human-centered cyber physical production systems. CIRP Journal of Manufacturing Science and Technology 22, 91–106.

[26] Deng, J., Wang, T., Wang, Z., Zhou, J., Cheng, L., 2022. Research on Event Logic Knowledge Graph Construction Method of Robot Transmission System Fault Diagnosis. IEEE Access 10, 17656–17673.

[27] Branden Chan, Stefan Schweter, Timo Möller. German's Next Language Model.

[28] Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., Stoyanov, V., 2019. Unsupervised Cross-lingual Representation Learning at Scale, 12 pp. http://arxiv.org/pdf/1911.02116v2.

[29] Nasar, Z., Jaffry, S.W., Malik, M.K., 2022. Named Entity Recognition and Relation Extraction. ACM Comput. Surv. 54 (1), 1–39.

Biography

Yuxi Wang, M. Sc. (*1995) is a research associate and PhD candidate at the Institute for Production Management, Technology and Machine Tools (PTW) at the Technical University of Darmstadt since 2021. Her research interests include the further development and introduction of natural language processing to digital shop floor management and problem-solving systems.

Beatriz Bretones Cassoli, M. Sc. (*1995) is a research associate and PhD candidate at the Institute for Production Management, Technology and Machine Tools (PTW) at the Technical University of Darmstadt since 2019. Her research investigates the use of graph-structured data and graph neural networks for predictive quality applications.

Prof. Dr.-Ing. Joachim Metternich (*1968) is head of the Production Management division at the Institute for Production Management, Technology and Machine Tools (PTW) at the Technical University of Darmstadt. He is the spokesman of the medium-sized businesses digital competence center Darmstadt; he has been president of the International Association of Learning Factories (IALF) from 2015 to 2023 and he is scientific partner of the Conference on Production Systems and Logistics (CPSL).