ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230642

Acquisition and Formalization of Tacit Knowledge for Value Chain Generation in Local Production Networks

Matthias Kerzel[b], Julia Markert[a;*], Emad Aghajanzadeh[b], Stephanie von Riegen[b], Lothar Hotz[b] and Pascal Krenz[a]

[a]Helmut-Schmidt-University, 22043 Hamburg, Germany
[b]Hamburger Informatik Technologie-Center, 22527 Hamburg, Germany
ORCiD ID: Matthias Kerzel https://orcid.org/0000-0002-1378-0435,
Julia Markert https://orcid.org/0000-0001-7370-7726, Emad Aghajanzadeh https://orcid.org/0000-0002-5381-1769,
Stephanie von Riegen https://orcid.org/0000-0001-8912-2423, Lothar Hotz https://orcid.org/0000-0003-3728-7634

Abstract. Interest in producing goods locally again has risen. This leads to new challenges for companies producing locally, especially since they are mostly small enterprises and do not always have resources to adapt Industry 4.0 technologies. Therefore, collaborating in networks can strengthen local production. We propose an online system with an underlying planning component that is supported by a large-scale language model to coordinate value chains within a network by utilizing the tacit production knowledge within the companies. Before any type of information processing can happen, however, the data – in this case, the tacit knowledge – needs to be acquired and formalized in such a way that is easy and quick, but also sufficient enough in detail and quality for the computer system. To this end, we conducted a study with 16 participants to simulate the collection of knowledge regarding the production of four pieces of furniture by having them describe simplified production steps. We analyze the results and show that the use of the collaborative system has a positive effect on the soundness of resulting production plans. In a second step, we utilize artificial intelligence methods to fill incomplete plans. Results and implications for future research are presented as well.

1 Introduction

Producing goods all over the globe with long value chains spanning multiple continents has been common in many industries for a few decades now and has served large companies and consumers well [9]. However, criticism in regard to this type of value creation has recently been getting louder, especially in the wake of the climate crisis and repeated supply chain problems leading to shortages of various products [22, 12]. As a result, some research has refocused on local production in smaller, networked, demand-based companies [12, 3]. Examples are the principles Urban and (Re-)Distributed Manufacturing that aim at a more sustainable production. Furthermore, the demand for more individualized products has increased in recent years [10], which also changes the requirements for a successful production that can satisfy customers' needs. Increasing product complexity further adds to the challenges of local production and of micro, small and medium-sized enterprises (MSME). The resulting raised exigencies for local stakeholders, e.g., a high product variety with small batch sizes, can be met better by companies collaborating in networks and combining their competences [24].

In this paper, we will therefore investigate the use of an online system as well as a matching mechanism based on the large-scale language model RoBERTA [14] to support the collaboration of micro, small and medium-sized enterprises (MSME). The purpose is to facilitate the acquisition and formalization of tacit production knowledge with the goal of creating value chains within a local production network. To evaluate the concept and applicability of the suggested technologies, a small-scale user study is presented. Figure 1 depicts a production plan from the user study and the resulting representation of production steps as planning actions, where matches between preconditions and effects are either established by a direct match or a match based on highest similarity.

The remainder of the paper is structured as follows: In Section 2 we motivate our approaches based on the requirements of local production networks. In Section 3 we survey the state of the art with regard to the extraction of formal planning representation from natural language and identify the current gap in this research. We present our web-based concept for the acquisition of tacit knowledge in Section 4 and detail its implementation in Section 5. We provide an evaluation based on a user study in Section 6 before we conclude our paper in Sections 7 and 8.

2 Motivation

The motivation for this paper emerged from a concept for the creation of cross-company value chains in networks of micro and small enterprises with very little to no digitization. Therefore, the concept is introduced shortly, followed by the problem and research question that we suggest a solution for in this paper.

2.1 Cross-Company Routing Planning

In order for companies to collaborate successfully in networks, some kind of concept needs to be developed and its implementation carefully planned and executed. In Markert et al. (2022) [15] a concept for creating value chains across company borders was introduced. The underlying idea, which we also follow, is based on the "precondition - action - postcondition" principle of PDDL (Planning Domain

* Corresponding Author. Email: julia.markert@hsu-hh.de.

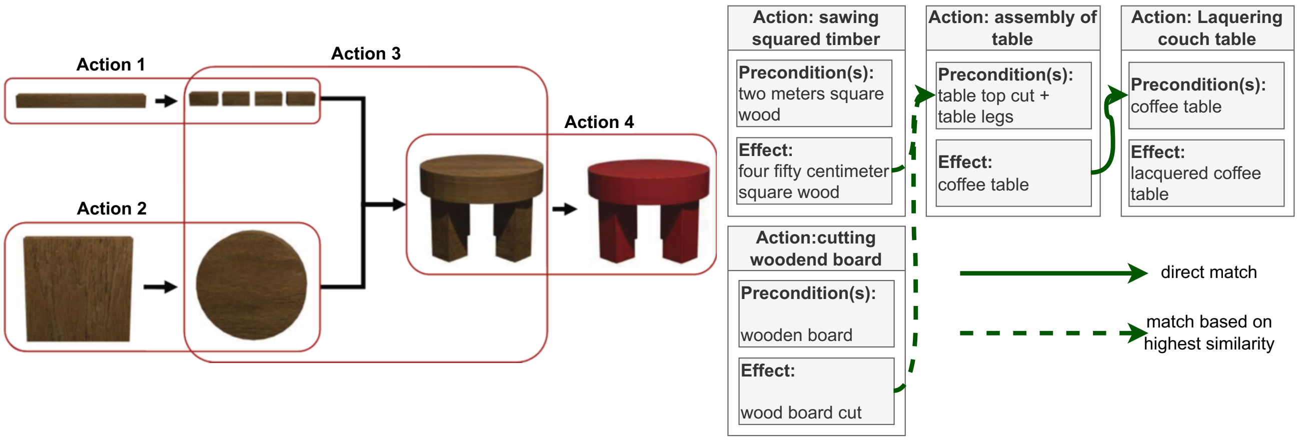

Figure 1. Example visualization of a production process and resulting planning actions.

Definition Language) [16]. It is suggested to use the tacit knowledge of workers for the routing planning of value chains cross-company without actually externalizing it. In order to do so, companies will be provided with the technical documentation of the product to be collaboratively produced in the network and will be asked to provide information to a central planning instance regarding what each individual company can contribute to the production of this specific product. In order to protect the companies' internal, specialized knowledge regarding their capabilities and machining processes, but also in order to allow for easier processing from an IT standpoint, the format for this information should be in accordance to PDDL. I.e., the companies will describe the product state that the workpiece needs to be in for them to start their production and the product state the workpiece will be in after they are done. Additionally, we suggest asking for an action name for each particular sequence of product states for easier identification.

The result should be a network plan with multiple action sequences that describe different ways to get from the raw material(s) to the finished products, with every action assigned to one or more companies that could execute it. Therefore, every time the product is ordered, the at that moment most suitable path through the network is determined and executed. For this to work, however, every product state (except for the overall start and finish) has to be the precondition for one action, and the effect (= "postcondition") for another, since otherwise there would be gaps in the process.

To summarize, a plan is a sequence of actions that lead to a target state. Actions are possible changes in the world. Preconditions define the conditions under which an action can be applied. An effect is used to describe the state of the world after an action has been applied. Value chains in this production approach consist of sequences of actions.

2.2 Research Question

The problem to be investigated in this paper pertains to the descriptions of the preconditions and effects of a product before and after its production steps, or - more precisely - to the acquisition and formalization of tacit knowledge in the form of the descriptions of the preconditions and effects. When asking multiple different people for a description of a product state, the result will likely be multiple different wordings for the same thing, different degrees of detailing or even just differences due to typing errors. Languages usually have multiple words to describe the same thing, there are regional and generational linguistic differences, words or abbreviations that are used only within a trade or even company etc. We hypothesize that these differences will also affect the state and action name descriptions considerably, causing them to rarely line up without further assistance. Thus, the input by the network partners needs to be processed in a way that formalizes the natural language descriptions of what can be done within each partnering enterprise to the point where it is clear to at least the producer before and after in the value chain, what the description means, and it is correctly processable by a computer system. Additionally, the collection or acquisition of said input should be easy and straightforward for network partners in order to require as little time as possible and to be as resistant to accidental mistakes as possible.

3 State of the Art

Planning is the task of finding a sequence of actions that leads an initial state to a desired (goal) state. Since these problems can get complicated quickly, AI techniques are often used to make the search more efficient. Nevertheless, for using AI in planning, we need to be able to represent the world's knowledge and reason based on it. Planning Domain Definition Language (PDDL) is a declarative language designed for this purpose [8]. A planning task in PDDL is specified using two files (Domain.pddl consisting of predicates and actions and Problem.pddl consisting of objects, initial states, and goal specification). Although PDDL offers an efficient means of automating the planning process, the need for encoding the knowledge into PDDL requires the expertise of an individual. However, in many applications, the lack of available experts to perform this task presents a challenge. This has led to a growing research interest in developing methods to derive the PDDL files from a high-level or even natural language.

[11] proposed a two-step approach for the dynamic generation of the PDDL domain and problem files from Web Ontology Language (OWL) files. The described models have been fully defined in the Extensible Markup Language (XML) Schema Definition Language (XSDL) and, based on the XML instance file, OWL files are gen-

erated. These generated OWL files are then used to generate the PDDL files. With [26] an approach was introduced, where Unified Modelling Language (UML) files are used as a basis for generating planning input. Requirements in UML are translated to solver-ready PDDL models in itSIMPLE3.0. Both papers show that automated generation of PDDL is possible; however, the generation is based on handcrafted XML or UML definitions which are required in the first place.

[17] described an approach for learning planning domain models directly from natural language (NL) descriptions of activity sequences in the form of process manuals. NL analysis is used to construct structured representations, from which formal representations of the action sequences are generated. The generated action sequences provide the necessary structured input for inducing a PDDL domain, using domain model acquisition technology. The approach uses a model based on reinforcement learning, which is a type of machine learning algorithm [25]. More specifically, they trained a deep Q network (DQN) [18] to derive linguistic annotations from text, like verbs to form action names or subjects and objects to form arguments, which is inspired from [7]. Having this knowledge then enabled them to employ a type of learning object-centred models (LOCM) method which is able to generate the PDDL domain file [5]. There are some disadvantages to the aforementioned ideas, though. Firstly, while this approach shows the applicability of deep learning methods in PDDL generation, it is based on the availability of consistent action sequence descriptions. However, the lack of these descriptions, and especially their inconsistency between different manufacturers, is one main motivation for our approach. Moreover, there is a lack of reports on the executability of the generated PDDL models using the approaches above. In other words, they have not tested their generated files on an actual planner and instead evaluated their models based on matching actions names and arguments. As a result, when attempting to use these models in our application, we were unable to generate sound PDDL files readable by the planners. Thirdly, since the method is based on deep learning, fine-tuning may be required before employing them, but in many domains, there is insufficient data to do so.

Recent advances in large language models, e.g., from the GPT family, have sparked research in automated NL to PDDL translation. However, the current state of the art shows that a fully automated translation is in many cases not successful. [27] report that GPT-3.5 is able to translate NL into planning goals; however, they identify issues with counting and spatial reasoning. [13] report the successful translation of natural language into PDDL problems, if the PDDL domain description, i.e., the description of actions and their preconditions and effects, is provided. However, this domain description is what needs to be formalized in terms of tacit knowledge in the first place.

In summary, existing approaches often rely on a consistent or already formalized representation of the planning domain or example action sequences to be fully successful. However, for the formalization of tacit knowledge from different manufacturers, we cannot rely on the availability of this information. Furthermore, a direct NL to PDDL translation without further guidance is prone to errors.

4 Requirements for the Acquisition of Tacit Knowledge for Production Planning

In order to acquire the information necessary for the creation of a value chain, the tacit knowledge of the network participants regarding what their company can manufacture or contribute to a certain product needs to be made usable. It is important here to note the distinction between "combining knowledge and making it usable" and "externalization of knowledge" according to Nonaka [19]. Externalization is an often complex process involving the creation of concepts using, e.g., analogies, metaphors, etc. and requiring people to ideally work closely together [20]. Nonaka and Takeuchi also describe it as "a quintessential knowledge-creation process" [20].

Such an elaborate and lengthy process is not suitable for quickly putting together very good (as there may not be a "best" solution) and dynamic value chains, which is exactly why the goal is not to externalize knowledge in the traditional sense, but just to combine and use tacit knowledge. The combination in this case is important because due to the networked nature of value chains, there are typically multiple experts from the network companies involved. Interestingly, "combination" is actually another mode of value creation according to Nonaka's model and usually transforms explicit knowledge to other / new explicit knowledge [19]. That is, however, a discussion for a different paper.

Relevant to this paper is that in order to use the knowledge from micro and small companies, it first needs to be collected from the workers or, if the company is big enough to have someone specifically for that role, the process planners. There are a few requirements regarding how the knowledge is collected:

1. **Data entry**: The system for the data entry must be easy and intuitive for the user. Making the user utilize a coding system, such as the one Opitz suggested [21], is not likely to be accepted initially by users, as it would require a lot of effort and time upfront before actual results are visible. Introducing a coding system later, however, for increased speed of data entry might be worth exploring.
2. **Time**: As typing information into a computer is not directly and visibly creating value, it is important to make this activity as fast as possible. While it is assumed that the process planners will not mind spending time on reviewing a technical drawing, they need to be able to quickly transfer their conclusions to the computer without typing long, drawn-out texts.
3. **Communication**: The Observatory of European SMEs found in a study that SMEs cooperate, but due in part to the immense time effort required for building trust and coordination, usually only with one to two partners [2]. In order to build bigger networks, it is, therefore, necessary to minimize the need for communication and dependencies. Network participants should be able to enter descriptions into the system at any time independently of others.
4. **Readability / interpretability**: The data entries from the users have to be read / interpreted by information technology. It is therefore vital to find the right combination of requirements for the data entry and requirements for the system, i.e., how strict are the rules of the data entry and how well does the system have to be able to compensate for things like, e.g., orthographic or syntax mistakes.
5. **Use of common terminology**: People from different backgrounds, countries (or even parts of the same country), generations, trades, etc. might use different terminology or words for the same product or process. Depending on the actual use case of the system, this may be almost unnoticeable or have a greater impact on the usability. Creating a dictionary with the most important terms or phrases or having a system that suggests alternative terms might be a possible solution to this issue.

In order to process the given information on "what each producer can contribute to the product" according to the concept introduced in [15] and briefly described in Section 2.1, the minimum information necessary is the following:

1. Basic information (such as company name, address and contact information)
2. Description of the start state of the product (or its parts), so the state before the company has contributed its labour to it
3. Description of the end state of the product (or its parts), so the states after the company has contributed its labor to it
4. Label for the state change (= manufacturing/production step)

Point 1 pertains to basic information about the producer that is independent of an order or product and does not have to match up with other network partners. Therefore, this paper discusses points 2, 3 and 4 only.

5 Concept and Implementation

In order to carry out the above concept, we have taken advantage of Django, which is a framework based on Python [6]. Django's layered architecture and flexibility make it a good choice in our application, wherein each part of the system might later be modified more easily as each part is independent of others. Each Django project contains three main layers, namely model, view, and template, which is called the MVT (Model View Template) design pattern. The template files form the layout of the user interface, where we provide various forms for users to register information, and the model files define the structure of the data being used in the system, i.e. actions, products, and plans. Moreover, the view files interact with the two other layers by accepting requests from the browser and sending them the appropriate responses. In our implementation, the translation from user input to PDDL files is done in view files.

5.1 User Interface for Knowledge Acquisition

The workflow of the system starts with someone, e.g. customers, designers, etc., who registers a specific product into the system. They provide the system with the product name, description, and some optional images. Following this, different manufacturers can log in, see all available products, and select one to contribute to. They will then be redirected to a page where they are asked to specify their contribution by entering an action. Similar to the definition of actions in PDDL, each action is identified by its name, precondition(s), and effect. In this first version of the system, we decided to accept only one effect per action, to simplify the user experience. The main innovation of the designed system lies in this part, wherein each manufacturer can see the other registered actions' preconditions and effects. Therefore, they are able to base their action description on other actions, which should result in significantly less inconsistent actions. Nevertheless, users are also able to submit their own precondition(s) or effect in case no existing one is satisfactory. Following this, once all manufacturers enter their actions, the administrator can log in and ask the system to generate PDDL files and the corresponding plan if that is possible. The system retrieves all actions from the database and automatically generates valid PDDL files, which are then used by the planner. It is worth mentioning, we did not use a learning-based method for this transformation, instead, we use a set of rules, which ensures that the generated files are sound and valid.

5.2 Matching Descriptions using Large Language Models

To repair plans, where the effect and preconditions of actions do not match due to different terminology by different network participants, we employ a similarity measure generated by a large language model. We use the pretrained *all-roberta-large-v1* [1] sentence transformer based on RoBERTA [14]. According to [23] this model is more robust toward out-of-domain input. The sentence-transformer translates sentences or text passages into 1024-dimensional vectors, so-called embeddings. These embeddings represent the semantic meaning of the encoded sentences, and the similarity of those embeddings can be used to estimate the similarity of the encoded input, in our case, the description of action, preconditions, and effect.

We employed the following algorithm: We assume that the last action of each plan is known, as it represents finishing the final product. We do regression planning from this final action, i.e., we look for (an) action(s) whose effects match the precondition(s) of the current last action in the plan. We first check if there are any actions whose effects exactly match the precondition(s) of the current action. If this is the case, we add this action to the plan. If no match can be established, we generate the RoBERTA embedding for the precondition and compute the cosine similarity of this embedding to the embedding of the effect of each action that is not yet part of the plan. We add the action with the most similar embedding to the plan. We continue with this process until only one action is left that is not yet part of the plan and add it without requiring a match. Note, as we are planning backwards from the final product, this action we add last to the plan will become the first action of the final plan.

Our approach creates a totally ordered plan, i.e., there is a clear order of actions, even if these actions could also be executed in parallel. E.g., it does not matter if we first prepare the tabletop or the table legs, as long as we have both prepared before assembling the table. The main reason for this is that, except for one occasion, parallelism was not modelled at all by the study participants. However, we take possible parallelism into account during the evaluation. As an example, Figure 1 shows action descriptions that do not match in terms of preconditions and effect. E.g., *"wood board cut"* and *"table top cut"* are not the same strings; however, among the options within this plan, this is the most likely match based on the RoBERTA embeddings, we can therefore suggest this as a possible match.

6 Evaluation of the Concept through a Study

In order to initially test the underlying principle of building value chains using the implementation described in Section 5 in contrast to collecting the information manually, we conducted a small-scale study. Study participants were given a consent form as well as the actual study form. The study was designed to be conducted remotely; therefore study participants received a full set of instructions that should enable the participation without further guidance, though a phone number was provided in case of additional questions. In the study form, participants were informed of the purpose of the study, asked to provide basic information on demographics and prior experience regarding the study topic (refer to Figure 3), they were given an example, as well as detailed instructions for the two-step study process and in the end were asked to fill out a short questionnaire on the usability of the system prototype. The study was given to participants in German or in English. Study participants were assigned ID-numbers to ensure that study results were not attached to names.

6.1 Study design

While the company information only needs to be entered once when registering the company to the system, entering the description of the product states is crucial to creating the value chains, but also rather complex. Therefore, we tested the entry of product states and action names in a two-step process.

For the "production processes", we used four simplified pieces of furniture, each with four simplified production steps. Each study participant was assigned one step per product, taking into consideration that it was not the same type of step for each product (e.g. assembly). They were shown the finished product, as well as the before and after images of only their assigned production step. Study participants were not given any further information regarding the other production steps. Figure 2 shows an example of such a step.

Using the images has the purpose of assigning the study participants "knowledge", that the other participants do not have. Because the study participants were not actual producers, they were provided with that information, since this study investigates only how the knowledge is described. In a real-life application, the producers/process planners intrinsically have the entire knowledge regarding the start and end state of a product in their production as well as all the production processes in between.

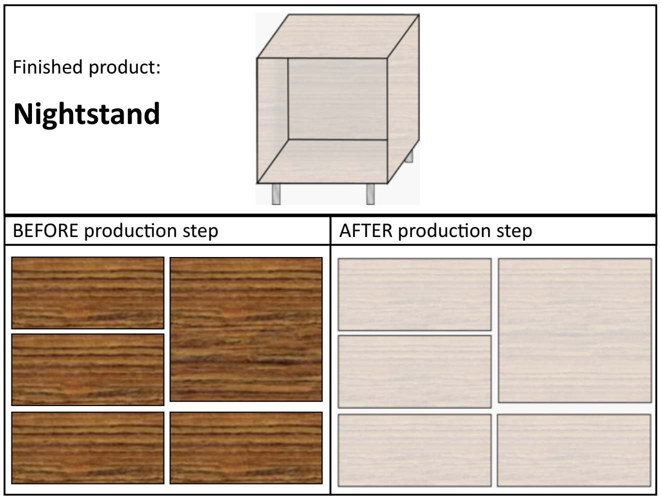

Figure 2. Example of a simplified product and process step through a before and after image.

Overall, we had 16 participants for the main study as well as 3 participants for a pre-study. The pre-study was used to improve and simplify the instructions and to identify and fix remaining bugs in the web application, but will not be discussed further in this paper. Instead, we will report on the results of the main study only. For the main study, participants were assigned into four groups and given one step for each product, meaning we received four descriptions by four different people for each "production step". Groups 1, 2 and 3 were German speakers and group 4 used the English study version.

6.1.1 Demographics and Background

In the demographics questionnaire, we asked four questions:

1. What is your age in years?
2. Do you have experience with automated planning tasks (field in computer science)?
3. Do you have experience with woodworking / carpentry / building things yourself?
4. This is the English (or German respectively) version of the study; how fluent are you in English (or German) according to your own estimate?

The results of questions 1,2 and 3 are shown in Figure 3. Question 4 was used to ensure that all study participants had a sufficient understanding of the study language because not only did they need to understand the instructions, but they also had to enter descriptions in natural language terms. Answers to this question demonstrated an overall good level of the respective language, with most being native speakers. The ages of participants were distributed between 18 and 65, with the majority being 35 years old or younger. 15 out of the 16 participants had little to no experience with automated planning tasks. There were no woodworking experts among the participants, three had "quite a bit" of experience, while 13 had little to none.

Figure 3. Demographics and entry questions of study participants.

6.1.2 Step 1: Manual entry of descriptions

In Step 1 of the study, participants were asked to describe the before and after states of their assigned steps and an action name "on paper" (or directly in a word/pdf file) without using the web application. They were instructed to use their own words. For the example in Figure 2 this could be (example, not taken from study results):

- **action name**: "applying lacquer to cut wooden panels"
- **before**: "cut to size wood panel pieces"
- **after**: "lacquered cut to size wood panel pieces"

We hypothesized that this would lead to mismatching preconditions and effects, as the study participants have no idea what terminology other participants will use or have used.

6.1.3 Step 2: Entry of descriptions via web application

In Step 2, study participants were asked to describe their production steps again, but this time using the prototype of a web application. The application should support the process by addressing the five requirements described in Section 4, by, e.g., shortening the time needed for the study (and in the future network) participants, but also by providing measures to synchronize the descriptions with the goal that this will lead to functioning plans.

To this end, we implemented the following functionalities in the web application (refer also to Section 5.1 :

1. Sign-in with participant's study ID
2. Main page with the products and their names as screen tiles
3. List of state descriptions (preconditions, effects) by all study participants from the same group to choose from
4. Possibility to enter new state descriptions
5. Entry of descriptions and names using natural language terms
6. List of the actions entered by the respective study participants

The list of state descriptions suggested to the user when entering a new action has several purposes. Choosing an item from a list is faster than entering a new description. Also, the work of describing a state is not done twice, instead, the work by the other study participants (or in the future network partners) is reused. Additionally, reusing existing descriptions automatically takes care of the synchronization process and the problem of common terminology. Communication is also minimized because users can automatically see the planning work others have already done for a specific product.

6.1.4 Usability Questionnaire

The last step for the study participants was the System Usability Scale (SUS) [4], a ten-question usability questionnaire. They were asked to answer on a five-point scale from 1 being "strongly disagree" to 5 being "strongly agree". Figure 4 shows the mean results of the questionnaire with their standard deviation. Furthermore, we asked for any additional comments or feedback in an open comment box. Overall, the answers suggest that the system was easy enough to use as can be inferred by questions 4 and 7, though our instructions were a little too long and complicated as seen in Question 2. In future studies, we might use a demonstration video instead, as was suggested by two study participants.

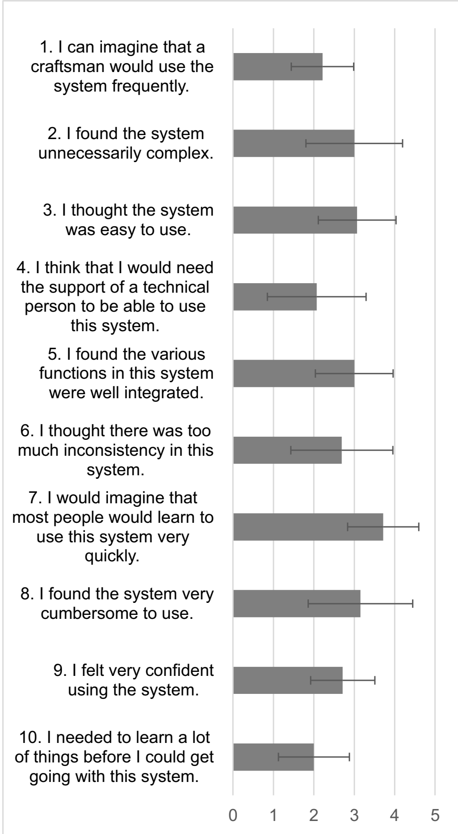

Figure 4. Mean results of the usability questionnaire with standard deviation (scale from 1 = strongly disagree to 5 = strongly agree).

6.2 Evaluation of the study results

First, we manually evaluate each entered action for correctness, i.e., we check if the action description, precondition and effect are understandable for a human reader, fit the described action from the questionnaire and are formally correct. In the manual part of the study, 98% of all actions are correct. In the second part of the study, using the web application, only 86% are correct; we attribute these errors to possible misclicks. Sometimes, actions were assigned to the wrong product and one action was formally incorrect, as the same precondition and effect were selected. We discard those plans that have missing or incorrect actions and continue with 12 (out of 16) plans, that have correct actions.

Next, we analyze the number of genuine matching preconditions and effects. As expected, there were no matches in the manual condition, as participants were not able to communicate with one another. However, we find some preconditions and effects, that are similar, such as (translated from German) *"couch table"* and *"not painted couch table"*. In the second part of the study, using the web application, only five such matches were established using the descriptions of the actions entered in the system (out of 3*3*4 = 36 possible matches), yielding a success rate of 13.89%. This unexpectedly low number might be attributed to different possible reasons. One cause could be that study participants did not actually use the system as intended by checking the list of pre-entered descriptions first when creating an action, but instead just copied and pasted their terms from the first step of the study into the web application. This may have been in part caused by the two-step study design, which is why in the future, only the web application should be used. Another possibility is that study participants did not deem the description by other participants accurate enough, or they did not understand the description and therefore used their own. Furthermore, we can attribute the relatively low number of direct matches to the unique perspective that each study participant had on the described action. Though study participants saw the final product as well as a before and an after picture of their product, they were not aware of what details might be relevant for other actions. Furthermore, as mentioned above, study participants might already have developed their own terminology during the manual part of the study, to which they adhered in the web application - this simulates well that craftsmen might be used to their own intra-company terminology. However, the successful matches show that such a system is, in principle, helping to establish a common terminology.

We have designed each product in the study in a way that some steps can be carried out in parallel and are put together in a later step by having two preconditions for an action, one for each separate part of the furniture to be assembled. Though this was described in the instructions with an example, only one study participant chose to model actions in this way. Most study participants chose to find a single precondition that represents all the required components, e.g., *"components of the nightstand"*. We attribute this to the fact that actions with multiple preconditions are less intuitive. This is going to be one issue we will address in future work.

Next, we use RoBERTA as a large language model to repair the plans, as described above. As the model is trained in English, we use automated translation from German to English for all actions, preconditions, and effects, which we verify with a manual check. We then automatically create totally ordered plans for each product based on either correct matches or the best match according to the large language model. We compare these totally ordered plans to the actual plans as intended in the study, which were partially ordered due to the parallel nature of the manufacturing process. As mentioned above, the majority of study participants chose not to model parallelism. Therefore, we rate an action as being placed correctly in a plan if it does not violate any of the ordering constraints of the partially ordered plan. I.e., we rate an action as correctly placed in the plan, if it happens after all other actions that are required for the action to be executed happen before in the plan and all other actions based on the effect of the action happen afterwards. According to this metric, we can establish a success rate of 66.67% for the possible matches.

6.3 Implications for future improvements of the web application

The results show that using the web application can help to establish a common terminology. Moreover, automated matching of preconditions and effects of actions can contribute to successful plans. In future work, we will focus on establishing an interactive process where actions entered into the system are immediately matched against existing actions and their preconditions and effects. In case no matches can be established, we can use the large-scale language model to generate suggestions for users. This will be especially useful once the complexity of plans and the number of actions will increase. To increase the usability of the system, we will develop more comprehensive instructions and produce different media, e.g., tutorial videos, to explain the web application more efficiently. In the next phase, we will conduct studies with actual craftsmen on actual products to evaluate our approach under the most realistic conditions.

Future development of the system will also encompass an intuitive visualization of value chain networks to give contributing manufacturers a comprehensive overview of the local production landscape. This could also help with guiding users more towards using existing descriptions instead of creating new ones. Furthermore, the automated detection of gaps in existing plans, in combination with a recommendation system for manufacturers, can help to automatically identify and mend gaps in value creation chains.

7 Outlook

In this paper, we present a prototype system and evaluation for acquiring and formalizing tacit knowledge for use in a planning system. However, there are several options we would like to address in future work:

To assess the practical benefits of the system in terms of saved cost and time, we plan to conduct a study with real manufacturers to compare the time and effort to existing workflows. For future studies, we want to establish a baseline by comparing the performance of our proposed system against existing solutions, as well as traditional approaches not aided by AI or computational systems. It is essential that future evaluations are based on real-world data and actual future users like manufacturers but also potential customers.

As our approach relies partially on LLMs, our future research will focus on an in-depth evaluation of different language models. While relevant benchmarks already exist in the literature, we will use these models in a very specific condition: We will use (in part) automatically translated texts that might also contain a lot of uncommon domain-specific vocabulary. It is currently an open research question how this affects performance. Moreover, for practical purposes, we not only plan to evaluate the performance in terms of correctness but also in terms of computational requirements and scalability.

Another important aspect is that the proposed system is intended to consider a multi-dimensional cost function of the value creation chain that not only takes into account monetary costs but also environmental issues like the minimization of transport routes and carbon dioxide footprint or the estimated production time. To this end, we will incorporate cost functions, which are supported by PDDL, into the planning process. The goal is to create an intuitive way for non-expert users to understand and control the trade-offs between different cost criteria.

Finally, we will assess the generalizability of our approach to other manufacturing or service domains like pharmaceuticals or event organization.

8 Summary and Conclusion

We present a novel approach to supporting local production networks with a planning-based online system. Different manufacturers can independently specify the production steps they want to contribute to, e.g., a furniture manufacturing process. These production steps are modelled as actions in a planning system that can automatically check production plans for completeness and identify weaknesses. A main challenge for such a system is the formalization of tacit knowledge of individual manufacturers into planning actions. While approaches exist that transform natural language into PDDL plans, they either rely on existing formalization or are prone to errors if they are solely based on large language models. To this end, we propose a system that utilizes two mechanisms: First, a collaborative web application enables users to see the action definitions of other users with regard to a certain product and use part of these definitions to define their own production steps. Secondly, with a matching algorithm based on a RoBERTA sentence transformer, we identify the most likely matches between preconditions and effects of planning actions to establish complete plans. We evaluate the system in a user study and can show that: a) Independently, users will not produce functioning plans, as they use their individual terminology. b) Using the web application, users do use existing action definitions as part of their own action definitions; however, often, they still use their terminology. c) A matching algorithm based on the similarity of embeddings generated by a sentence-transformer can establish functioning plans. The presented paper contributes to the research on the application of artificial intelligence in local production networks and will be the basis for the development of an actual platform in a metropol area.

Acknowledgements

This research is funded by dtec.bw - Digitalization and Technology Research Center of the Bundeswehr [project: Production Next Door]. dtec.bw is funded by the European Union – NextGenerationEU.

References

[1] all-roberta-large-v1. https://huggingface.co/sentence-transformers/all-roberta-large-v1. Accessed: 2023-04-26.

[2] *KMU und Kooperationen*, volume Nr. 5 of *Beobachtungsnetz der europäischen KMU (Observatory of European SMEs), 2003*, Amt für Amtliche Veröff. der Europ. Gemeinschaften (Office for Official Publications of the European Communities), Luxemburg, 2004.

[3] *Produktionsarbeit der Zukunft - Industrie 4.0: Studie*, Fraunhofer-Verl., Stuttgart, 2013.

[4] John Brooke et al., 'Sus-a quick and dirty usability scale', *Usability evaluation in industry*, **189**(194), 4–7, (1996).

[5] Stephen N Cresswell, Thomas L McCluskey, and Margaret M West, 'Acquiring planning domain models using locm', *The Knowledge Engineering Review*, **28**(2), 195–213, (2013).

[6] Django Software Foundation. Django. https://djangoproject.com. Accessed: 2023-05-05.

[7] Wenfeng Feng, Hankz Hankui Zhuo, and Subbarao Kambhampati, 'Extracting action sequences from texts based on deep reinforcement learning', *arXiv preprint arXiv:1803.02632*, (2018).

[8] Maria Fox and Derek Long, 'Pddl2. 1: An extension to pddl for expressing temporal planning domains', *Journal of artificial intelligence research*, **20**, 61–124, (2003).

[9] Liena Kano, Eric W. K. Tsang, and Henry Wai-chung Yeung, 'Global value chains: A review of the multi-disciplinary literature', *Journal of International Business Studies*, **51**(4), 577–622, (2020).

[10] Cindy Kohtala, 'Addressing sustainability in research on distributed production: an integrated literature review', *Journal of Cleaner Production*, **106**, 654–668, (2015).

[11] Zeid Kootbally, Craig Schlenoff, Christopher Lawler, Thomas Kramer, and Satyandra K Gupta, 'Towards robust assembly with knowledge representation for the planning domain definition language (pddl)', *Robotics and Computer-Integrated Manufacturing*, **33**, 42–55, (2015).

[12] Mats Larsson, *Circular business models: Developing a sustainable future*, Palgrave Macmillan, Cham, 2018.

[13] Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone, 'Llm+ p: Empowering large language models with optimal planning proficiency', *arXiv preprint arXiv:2304.11477*, (2023).

[14] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov, 'Roberta: A robustly optimized BERT pretraining approach', *CoRR*, **abs/1907.11692**, (2019).

[15] Julia Markert, Dominik Saubke, Pascal Krenz, and Lothar Hotz, 'Cross-company routing planning: Determining value chains in a dynamic production network through a decentralized approach', (2022).

[16] Drew McDermott, Malik Ghallab, Adele Howe, Craig Knoblock, Ashwin Ram, Manuela Veloso, Daniel Weld, and David Wilkins, 'Pddl-the planning domain definition language', (1998).

[17] Shivam Miglani and Neil Yorke-Smith, 'Nltopddl: One-shot learning of pddl models from natural language process manuals', in *Proc. of the ICAPS Workshop on Knowledge Engineering for Planning and Scheduling (KEPS). ICAPS*, (2020).

[18] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al., 'Human-level control through deep reinforcement learning', *nature*, **518**(7540), 529–533, (2015).

[19] Ikujirō Nonaka, 'A dynamic theory of organizational knowledge creation', *Organization Science*, **5**(1), 14–37, (1994).

[20] Ikujirō Nonaka and Hirotaka Takeuchi, *The knowledge-creating company: How Japanese companies create the dynamics of innovation*, Oxford University Press, New York, 1995.

[21] Herwart Opitz, Wilfried Olbrich, and Gerhard Steinmetz, *Automatische Arbeitsplanerstellung*, volume 2296 of *Forschungsberichte des Landes Nordrhein-Westfalen*, Westdt. Verl., Opladen, 1972.

[22] Sudhir Rama Murthy, Steve Evans, and Joseph Sarkis, 'The four freedoms-of-movement and distributed manufacturing', in *Sustainable Development Goals and Sustainable Supply Chains in the Post-global Economy*, eds., Natalia Yakovleva, Regina Frei, and Sudhir Rama Murthy, volume 7 of *Springer eBooks Earth and Environmental Science*, 47–66, Springer, Cham, (2019).

[23] Nima Sadri, 'Evaluating dense passage retrieval using transformers', *arXiv preprint arXiv:2208.06959*, (2022).

[24] Günther Schuh, Kai Millarg, and Åsa Göransson, *Virtuelle Fabrik: Neue Marktchancen durch dynamische Netzwerke*, Hanser, München, 1998.

[25] Richard S Sutton and Andrew G Barto, *Reinforcement learning: An introduction*, MIT press, 2018.

[26] Tiago Stegun Vaquero, José Reinaldo Silva, Marcelo Ferreira, Flavio Tonidandel, and J Christopher Beck, 'From requirements and analysis to pddl in itsimple3. 0', *Proceedings of the Third International Competition on Knowledge Engineering for Planning and Scheduling, ICAPS 2009*, 54–61, (2009).

[27] Yaqi Xie, Chen Yu, Tongyao Zhu, Jinbin Bai, Ze Gong, and Harold Soh, 'Translating natural language to planning goals with large-language models', *arXiv preprint arXiv:2302.05128*, (2023).