Article

An Empirical Evaluation of Prediction by Partial Matching in Assembly Assistance Systems

Arpad Gellert [1,*], Stefan-Alexandru Precup [1], Bogdan-Constantin Pirvu [2], Ugo Fiore [3], Constantin-Bala Zamfirescu [1] and Francesco Palmieri [4]

1 Computer Science and Electrical Engineering Department, Lucian Blaga University of Sibiu, 550025 Sibiu, Romania; stefan.precup@ulbsibiu.ro (S.-A.P.); constantin.zamfirescu@ulbsibiu.ro (C.-B.Z.)
2 Industrial Engineering and Management Department, Lucian Blaga University of Sibiu, 550025 Sibiu, Romania; bogdan.pirvu@ulbsibiu.ro
3 Department of Management Studies and Quantitative Methods, Parthenope University, 80133 Napoli, Italy; ugo.fiore@uniparthenope.it
4 Department of Computer Science, University of Salerno, 84084 Salerno, Italy; fpalmieri@unisa.it
* Correspondence: arpad.gellert@ulbsibiu.ro

Abstract: Industrial assistive systems result from a multidisciplinary effort that integrates IoT (and Industrial IoT), Cognetics, and Artificial Intelligence. This paper evaluates the Prediction by Partial Matching algorithm as a component of an assembly assistance system that supports factory workers, by providing choices for the next manufacturing step. The evaluation of the proposed method was performed on datasets collected within an experiment involving trainees and experienced workers. The goal is to find out which method best suits the datasets in order to be integrated afterwards into our context-aware assistance system. The obtained results show that the Prediction by Partial Matching method presents a significant improvement with respect to the existing Markov predictors.

1. Introduction

The Smart Factory vision in Industry 4.0 is a complex system where artefacts collaborate with people, facilitated by a complex exchange of data among all interactants, both in the physical and in the digital world. In this data-driven collaboration environment, human assistance systems play a major role by providing the most suitable information at the right time, as specified in the Operator 4.0 concept [1,2].

In Europe, the jobs involving manual work constitute the second-largest category within the manufacturing sector [3]. Assistive systems are nowadays important components of the manufacturing industry, supporting humans to better deal with the increasing complexity of products and operating procedures, as well. Assistance systems support and guide workers to learn and improve their skills, safeguarding their security, while being unobtrusive, but ensuring that the operators are always in control.

To support the workers' cognitive effort during manual assembly processes, appropriate, dynamic, context-dependent instructions should be provided to the worker in a timely and ergonomic way. In addition, mistake detection modules that check whether a manufacturing process is executed and completed as specified reduce the need for rework [4]. Modern factories are relying more and more on such systems to help workers and eventually increase efficiency. For complex tasks that require skilled workers, computer-aided assistive systems are often used to provide guidance in the manufacturing process. Some production processes must be completed in a fixed sequence of subtasks, while in other processes the same result may be accomplished through different orderings, and there may not exist an optimal order, as different procedures may fit various specific characteristics

of the worker. Sensors are widely used to detect body position and motion, to recognize facial expressions, and to identify objects. Such information, recorded within an assembly station, can be used to recognize the current state of the manufacturing process and to recommend possible next assembly steps. In essence, it can be said that assistive systems represent a multidisciplinary effort, involving IoT (and Industrial IoT), cognitive aspects, and Artificial Intelligence.

In the previous works [5–8], several context-based predictors were investigated. These predictors provide assembly instructions based on the current and some past assembly states. Two-level context-based predictors, Markov models and Long Short-Term Memory (LSTM), have been evaluated and the results have shown deficiencies in terms of coverage. A more detailed description of these methods is given in Section 2. As the goal is to integrate the most efficient predictor in the assembly assistance system to support the workers with choices for the next assembly step, other methods will be further investigated.

This paper explores the predictor able to provide multiple choices for the next assembly step. By combining multiple Markov models of different orders, the Prediction by Partial Matching (PPM) can better identify manufacturing patterns. Thus, we expect that the previously mentioned assembly modeling methods will be outperformed by the PPM. For the case study, a customizable modular tablet was chosen as the target product to analyze the efficiency of the PPM in predicting the next assembly step. Prediction rate, coverage, and prediction accuracy were used as efficiency metrics. The method exploits the last assembly step(s) as input information, as well as characteristics of the human worker, such as the gender, use of eyeglasses, height, and the sleep quality in the previous night. The evaluation of the proposed method was realized using datasets collected from an experiment based on a case study, in which 68 inexperienced workers and 111 experienced factory workers performed unrestricted assemblies of the target product. Thus, the proposed predictors can learn correlations between assembly styles and characteristics of the human workers. These correlations are exploited afterwards in order to suggest the next assembly step under different conditions. The final decision support system can be very useful to inexperienced workers in an effective way, by replacing a costly training process. Even experienced workers can benefit from using it, especially in long manufacturing processes in which monotony and tiredness can otherwise lead to mistakes.

The rest of this paper is organized as follows: Section 2 presents related work, Section 3 describes the proposed prediction methods used to recommend the next assembly step, Section 4 discusses the evaluation, whereas Section 5 concludes the paper and proposes some further work directions.

2. Related Work

Since the pioneering work of Yamada et al. [9], a consistent flow of research on skill assistance systems for workers has been conducted. A review of smart manufacturing systems is provided in [10]. A human–machine centric assembly station is proposed by Rojas et al. [11]. In that work, the focus is on a case study in a mini-factory laboratory, equipped with devices for manual assembly as well as devices used for automated or hybrid assembly. The manual workstations had flexible plug-in systems of tubular frames and tables, equipped with electric screwdrivers and grab containers. Other elements were the lean Kanban flow racks used to apply material commission. The laboratory had software systems and several robots which could be used for automated assembly demonstrations. The students who participated in the case study simulated the manual assembly of pneumatic cylinders.

Elkomy et al. [12] presented the ABBAS biosensor-centric assistive system and investigated the feasibility of using biodata. Funk et al. [13] discussed the requirements for providing cognitive support at the workplace. Bertram et al. [14] analysed the intelligent workstations available in both research and industry applications. Several implementations were presented, including the Bosch Rexroth's "ActiveAssist" station and "Plant@Hand" from the Fraunhofer Institute for Computer Graphics Research. The following aspects were

considered important in a workstation: assisted work instructions, automated generation of work plans, detection and recognition of tools, detection, and recognition of worker contexts and assembly processes, flexible and adaptable integration in production and automated learning ability.

Mueller et al. [15] presented the development of an assembly assistance platform especially dedicated to rework stations. It comprised a planning environment, where processes and parameters are defined, and a control system. In contrast to the present work, the input to the planning module is defined manually.

Gorecky et al. [16] presented a virtual training technique intended to improve production transparency for the human worker and to ensure faster adaptation to new situations. The trainees can acquire relevant knowledge about the involved components, as well as assembly positions, modalities, and sequences. There are three stages in the learning process: an easy mode for familiarization, a medium mode and, finally, a hard mode where the appropriate components must be actively chosen. The knowledge is further reinforced through specific games. The automated training content is generated by an interoperable information interface. In [17], the impact of virtual training on the workers' learning process was presented. The work concluded that a virtual simulation performed before the real manufacturing increased the workers' efficiency. In [18], gamification was applied to get the worker fully immersed in the production activities.

SOPHOS-MS is a human–machine collaboration solution, relying on Augmented Reality (AR), designed to provide visual instructions to humans, as well as containing an intelligent personal digital assistant for voice interaction between the human and the expert system. It offers operators real-time feedback (i.e., visual and voice) on tasks, procedures to enable the safe operation of manufacturing systems, as well as a training approach. The evaluation of the proposed solution, in contrast with the classical training of operators, was conducted in order to set up a CNC machine for a given component, proving better results for operators trained by SOPHOS-MS [19].

Lai et al. [20] developed an assembly training system providing on-site augmented reality instructions (i.e., text, video, and 3D animations) to reduce time and errors made by human operators. The generation of augmented reality instructions relies on deep learning networks (i.e., Region-Based Convolutional Neural Network) trained on synthetic data to provide the appropriate information when the user engages with a specific tool to perform an assembly task. The study revealed that the system reduced the duration and also the assembly errors by approximately 33%.

Loskyll et al. [21] proposed a context-based orchestration framework composed of three layers: service registration, service discovery and selection, as well as service orchestration. Ontologies were modeled for semantic reasoning in the discovery and selection of services and were used to describe the web services provided by field devices. The service with the highest score, weighted among several matching criteria, was selected. The process was then decomposed in atomic processes and the corresponding services were invoked for the resulting composite process.

In [5], a two-level contextual predictor was used to suggest the next assembly steps. The first level of such a predictor consisted in a left-shift register which contained the last assembly states. The second level was a prediction table, storing pairs of state-patterns and their associated next states. The left-shift state register selected an entry from the prediction table. Then, the state from the selected entry was provided as the predicted one. Another variant of the two-level contextual predictor extended each assembly state with an automaton which could be in stable or in unstable substate. The predictor could learn a different state for a certain pattern, only if its current state was unstable, otherwise it just switched from stable to unstable and kept the current state. Unfortunately, this scheme provided insignificant improvement, and because it used supplementary information and additional steps in the prediction process, it was considered less efficient than the scheme without automata. The Markov predictor presented in [6,7] is another two-level predictor which can store multiple next states, together with their number of occurrences

for each pattern. The state with the highest number of occurrences is extracted from the prediction table entry selected with the left-shift state register and is then provided as the predicted one. Such a predictor can provide multiple next assembly choices, but for time-critical decisions it can be configured to return the most probable state. Article [8] presents the use of LSTM in modeling assembly processes. The LSTM network uses the human characteristics and the current assembly state to make a prediction of the next assembly state. The implementation of the recurrent network was done through TensorFlow and Keras and is composed of several layers: an input layer, two LSTM layers and an output layer. Several tests have been performed to finetune the hyperparameters. Although the proposed network has a lower accuracy than the Markov predictor, it can better adapt to new scenarios, being able to correctly predict over twice as many assembly steps as the Markov predictor.

3. Prediction-Based Assembly Support System

This section provides a detailed description of the assembly assistance system proto-type. The focus is on its prediction module, which is essential for assistive capabilities. The PPM algorithm is used to provide choices for the assembly step within assisted manufac-turing processes.

3.1. Assembly Assistance System

The goal of the assembly assistance system (Figure 1) is to support operators to correctly learn the manual assembly process for a customizable tablet, without any human intervention (i.e., trainer).
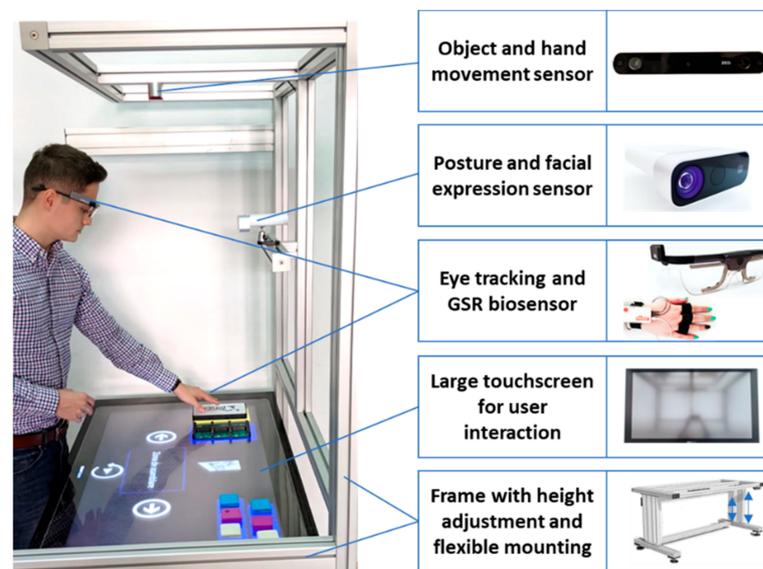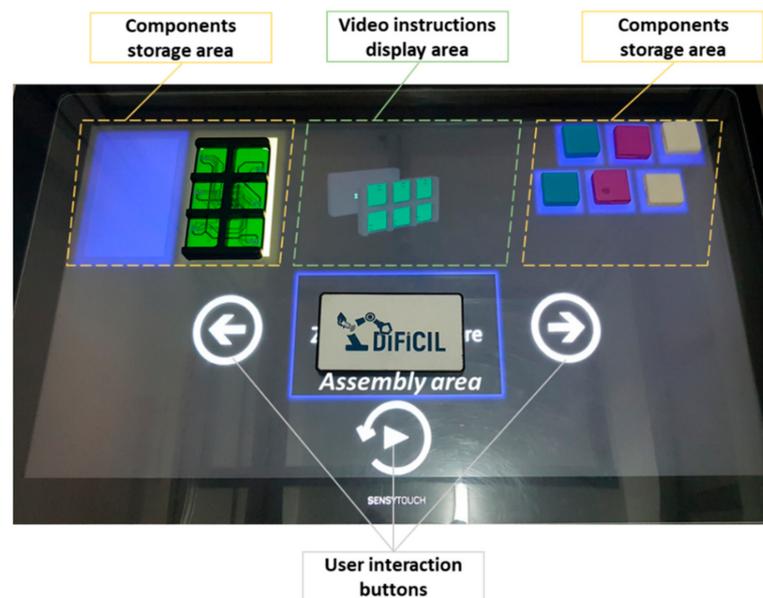

Figure 1. Assembly assistance system.

From a hardware point of view, the assembly assistance system is composed of five main components. The aluminum frame has embedded electrical motors that enable the height adjustment of the assembly table. The frame's aluminum profiles enable easy and flexible mounting of other relevant devices on the frame. The Sensytouch ST43 SLIM device is used to display information, to enable user interaction, and to run the training application and the required services (presented below). Because it is embedded in the frame as a tabletop and because it has a protective cover on the touchscreen, all the available surface can be used as working space (assembly and storage area, with the required assembly components laid on it). Its main characteristics are presented in Table 1.

Table 1. Characteristics of Sensytouch ST43 SLIM.

Component	Characteristics
Display	43-inch 4K touchscreen
CPU	Intel i7-7700
GPU	NVIDIA GeForce GTX 1060
RAM	16 GB
SSD	250 GB
Operating System	Windows 10

The sensor for posture and facial expression detection is placed in front of the user. For this, a Microsoft Kinect Azure is utilized. Tobii's Pro 2 glasses are worn for eye tracking and gaze analysis, while a Shimmer sensor is mounted on a finger for galvanic skin response (GSR) readings. The sensor for object and hand movement is mounted on the upper part, above the table and hands. For this, Stereolabs' ZED 3D camera is used.

The application's user interface is exemplified in Figure 2, highlighting the main areas for instruction, interaction (i.e., buttons for start, repeat, previous, and next), storage, as well as the location where to store the assembled components after each instruction.

Figure 2. User interface displayed on the large touchscreen.

From a software perspective, the assembly assistance system is provided with micro-services to enhance user experience. All micro-services communicate via the gRPC framework. Additionally, each micro-service has its own Health Check and Service Discovery mechanisms. These mechanisms allow us to select which services will collaborate at any given time.

Next, the micro-services which are relevant from the perspective of assembly prediction and user behaviour are presented. The height adjustment of the assembly station can be done manually, by pressing physical buttons on the station, or from software. The experiment revealed that the height of the subjects is one of the major factors that influence the assembly process. The depth camera streaming service allows the clients to control the depth camera as if they were connected to it. It exposes all the camera capabilities (RGB, depth, point cloud, etc.). The object detection service detects the position of the objects in each image. The object position service combines object detection and depth camera microservices to establish the 3D position of objects relative to the camera. Additionally, it detects if the objects have been inserted correctly in their slots. In case of a wrong step, it

will prompt the user to undo the action. This feature acts as a safeguard for the prediction service, since it cannot detect the incorrect assemblies, nor should this be its responsibility. In the future, this feature will be extracted in its own service. The emotion detection service relies on face mimics. This service detects the emotion of the user based on a picture of his face. Ongoing work is to be done to estimate emotion based on the data fusion from both mimic and GSR. The human characteristics collector service collects human characteristics such as age and gender. The mood can be identified with the aid of "emotion detection based on face mimics" microservice. These characteristics aid in the prediction process due to factors or preferences representative for a segment of the population. The prediction service (running on the Sensytouch device described earlier) receives information collected from the other services through an aggregator and, based on its various algorithms, it should return the next recommended assembly step. Its role is to guide the trainees during their training stage and optionally to be used as a detector of incorrect assembling for experienced workers.

These services change the way the user interacts with the assembly station. They reduce the probability for the user to make a mistake and introduce better recommendations for each individual user. Due to their nature, all these services are plug and play, enabling the supervisor to run only the services he deems necessary during the training. Thus, the services allow for great interoperability.

The tablet is made up of maximum eight components (Figure 3): one screen, one mainboard, and six modules which can be speakers (white), flashlights (purple) or batteries (blue). The customization consists in the selection of quantities for module types. The mainboard is the base component on which the other pieces are assembled. All the other components have a bit associated in the code, which describes a certain state of the tablet. If a component is correctly assembled, its corresponding bit is "1". Otherwise, if a component is wrongly assembled or not yet mounted, its corresponding bit is "0". Thus, a seven-bit integer represents the assembly completion of the tablet, 127 representing a fully assembled tablet and 0 a disassembled tablet. The first bit of the representation is for the screen, the next three are for the top row, and the last three for the bottom row.

Figure 3. Binary representation of the tablet's state.

Due to this binary codification, the system can detect which component has been mounted, by comparing the codification of two consecutive states. Moreover, by comparing the current state with the predicted next state, it is possible to determine which component should be next assembled.

3.2. Providing Choices for the Next Assembly Step through Prediction by Partial Matching

For the next assembly step prediction, the PPM algorithm is applied, which is, in fact, a hybrid method composed of Markov chains of different order. It was applied in data compression, but it was used also in other applications relying on prediction, like web prefetching [22], branch prediction in microprocessors [23], etc. In a Markov chain of order R, the next state probability can be defined as $P[q_t|q_{t-1}, \ldots, q_1] = P[q_t|q_{t-1}, \ldots, q_{t-R}]$, where q_t is the state at time t. As presented in [6,7], the prediction with Markov models can be performed using a table trained with past data, which stores pairs of patterns of a certain length and the corresponding next states, together with their occurrence frequencies. Figure 4 presents the structure of a Markov predictor of order R. The patterns in this work are composed of human worker characteristics $(C_1, \ldots, C_4)$ and assembly sequences $(q_{t-R}, \ldots, q_{t-1})$. The next state can be predicted if the current context is found in the pattern field of the table and then the state q_t with the highest frequency from that entry will be the prediction.

Pattern	States
⋮	⋮
$C_1, \ldots, C_4; q_{t-R}, \ldots, q_{t-1}$	q_t
⋮	⋮

Figure 4. Rth order Markov predictor.

A PPM of order R tries to provide a prediction with the Markov predictor of order R and, if it can do that, its prediction is returned by the PPM. Otherwise, the order of the Markov predictor is iteratively decremented until a prediction can be provided. If the Markov predictor of order 1 cannot issue a prediction, then the PPM itself is not able to do that. The PPM's prediction mechanism is depicted in Figure 5. For the PPM implementation, the Markov predictor with padding (presented in [7]) is used. The Markov predictor was enhanced to also use the workers' characteristics. Upon instantiation of the PPM algorithm, the order R should be provided. For each order, starting from R and moving down to 1, a Markov predictor is created. After all the R Markov predictors are created, they are sorted in descending order. When the prediction algorithm is called, it will iterate sequentially through the Markov predictors to find one which has a match for the assembly pattern, and so it can predict. If the Markov predictor of order R has no match, then the next Markov predictor from the sorted list is checked. The first Markov model that has knowledge of the assembly pattern is the one that will make the final prediction. If no Markov model can find a matching sequence, then the PPM algorithm is going to return −1, meaning that it cannot make a prediction. As far as we know, PPM has not been used for next assembly step prediction before. The novelty of this method consists in the combination of different order Markov predictors, which ensures that matches for the occurring assembly patterns can be found easier.

There are over 5000 possible ways of assembling the tablet and 4 characteristics that can define the behaviour of a worker, each one with 2 possible outcomes. Thus, the tablet could have over 80,000 unique assembling possibilities. Because of the high diversity of assembly patterns, the current context cannot always be found in the prediction table of the prediction scheme presented above, which negatively affects the prediction rate. Therefore, an enhanced scheme was considered, which explores neighboring characteristics of the user, whenever the current context (with the actual user characteristics) does not have an exact match. This approach would be practicable even if several additional characteristics were considered.
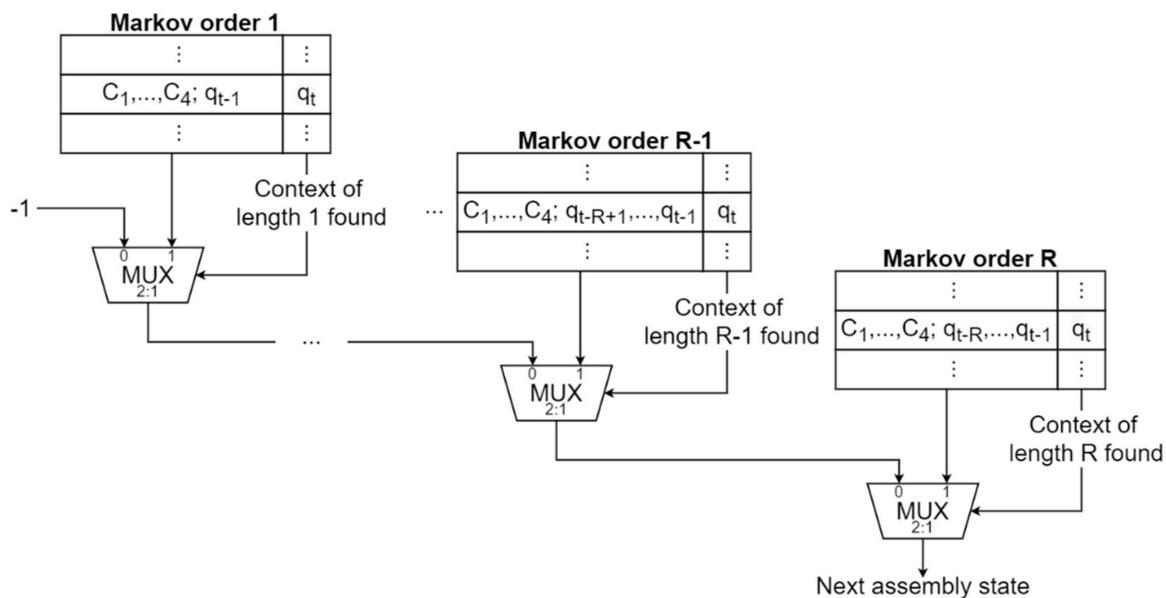
Figure 5. The prediction mechanism of the PPM.

Neighboring context exploration involves changing one characteristic of the worker at a time. A single trait of the worker is changed sequentially at a time, after which making a prediction is attempted. For example, if the user is a tall male wearing eyeglasses that slept well the previous night, and there is no match for this combination, predictions will be made by varying his characteristics (one at a time), obtaining four different neighboring states. In one of them, the gender is the changed variable, so that a prediction for a tall female wearing eyeglasses that slept well the previous night is made. Afterwards, the variable for height will be changed, then the one related to whether the worker slept well, and lastly the one related to wearing glasses. This approach might yield up to 4 predictions, given that the model has knowledge about the neighbors. The step that was predicted the most from the neighboring states will be considered the next assembly step (majority voting). Ties are resolved by selecting the most frequent prediction, having the lower index in the prediction list. If the model has no match for the assembly state paired with the user's characteristics, nor with neighboring characteristics, then no predictions can be made.

4. Evaluation

4.1. Experimental Methodology

The experiment involved two basic groups, 68 second year BSc students and 111 factory workers, who assembled freely, without guidance, the customizable modular tablet composed of eight components (as described in Section 3.1). The following setup was used for the experiment: a top-mounted camera that recorded the assembly steps, a table that had the working zone marked with red tape and two images of the tablet (front view and back view) on the left of the area. The tablet's components had their positions marked so that everyone encountered the same setup. Additionally, two laptops were used. One was connected to the camera and one was used for remote access.

Upon entering the room, the participant was provided with an ID. Since minimal interaction with the subjects was desired, the experiment was controlled from another room. The participants could only listen to a recorded short voice message, which was carefully recorded to prevent any transmission of feelings or emotions. The instructions in the message stated that the tablet should be assembled as indicated in the images and that they were required to use all the components in the assembly process. After the voice message finished playing, the participants started the assembly process. When the subjects considered the product assembled, they made an announcement, so the recording was

stopped and they were taken to another room where they filled in a questionnaire. The ID given at the start of the experiment was used in the questionnaire to help link the answers to the recording. There were general questions regarding height, age, gender, dominant hand, highest level of education completed and if they were eyeglass wearers. Other questions were for self-evaluation: "were you hungry during the experiment?", "do you have any prior experience in product assembly?", "what was your stress level before the experiment?", "how would you describe the state you found yourself in during the experiment (at the beginning, during and at the end of the experiment)", "are you under the influence of any drugs that might influence your level of concentration?", and "how would you describe the sleep quality of the previous night?" After answering these questions, they were tasked to complete a perception test.

After all the assembly steps were encoded, each assembly was ranked with a score that represented the degree of completeness of the tablet. Each correctly placed component was worth 1 point of the final score, with a total maximum of 7. As an example, the score of a subject who assembled all the components, except the screen, was 6/7. Additionally, each assembly had its duration in seconds recorded.

Two datasets were obtained through the experiment, further denoted as "Trainees" and "Workers". The "Trainees" dataset is composed only of students' assemblies, while the "Workers" one contains assemblies made by factory workers.

4.2. Experimental Results

This subsection presents the results of the proposed predictor, which will also be compared with other existing prediction methods. The aim is to compare both the capacity of learning the entire dataset and the capacity to adapt to new assembly scenarios. Three metrics have been chosen to evaluate the performance of the prediction algorithms: prediction rate, accuracy and coverage. The prediction rate measures how many times the algorithm was able to make a prediction. The prediction rate is computed in relation to the size of the testing dataset. The accuracy measures how many of the predictions made were correct and the coverage considers the correct predictions in relation to the whole testing dataset.

In the tables presented below, for each metric there are two columns "100/100" and "75/25". These two columns refer to how the algorithms were evaluated: "100/100" indicates that both the training and testing datasets were generated using 100% of the dataset, while "75/25" means that 75% of the dataset was used on the training of the model and the rest of 25% was considered for testing.

Table 2 presents the three metrics for the PPM algorithm of orders 1 to 7, on the "Trainees" dataset. The prediction rate remains constant throughout the orders for both testing methods. For the "100/100" testing method, the accuracy and coverage increase in small amounts up to order 3, where maximum percentages are achieved. Due to higher orders, the sequence can be more tailored to the worker, thus an increase of the accuracy and coverage can be observed. For new data, it seems that the metrics remain constant throughout all the orders.

Table 2. Evaluation of the PPM algorithm on the "Trainees" dataset.

PPM Order	Prediction Rate (%)		Accuracy (%)		Coverage (%)	
	100/100	75/25	100/100	75/25	100/100	75/25
1	95.62	54.46	80.05	61.82	76.55	33.66
2	95.62	54.46	82.21	61.82	78.61	33.66
3	95.62	54.46	82.48	61.82	78.87	33.66
4	95.62	54.46	82.48	61.82	78.87	33.66
5	95.62	54.46	82.48	61.82	78.87	33.66
6	95.62	54.46	82.48	61.82	78.87	33.66
7	95.62	54.46	82.48	61.82	78.87	33.66

As presented in Section 3.2, the PPM predictor was improved by enabling it to search in all the neighboring states (when it is necessary) for a possible assembly. The PPM with neighboring is further denoted PPMN. Table 3 presents the results of the PPMN on the "Trainees" dataset. On known data (with the "100/100" testing method), a small increase can be observed in terms of coverage, a 2% increase in the prediction rate, while the accuracy was slightly lower than that of the PPM without neighboring states. When it comes to new data ("75/25" testing method), the coverage increased by over 10% compared to the PPM without neighboring search and the prediction rate by over 20%. Although there is a slightly lower accuracy, the use of this enhanced algorithm is preferred.

Table 3. Evaluation of the PPMN on the "Trainees" dataset.

PPMN Order	Prediction Rate (%)		Accuracy (%)		Coverage (%)	
	100/100	75/25	100/100	75/25	100/100	75/25
1	97.68	77.23	78.89	57.69	77.06	44.55
2	97.68	77.23	81	57.69	79.12	44.55
3	97.68	77.23	81.27	57.69	79.38	44.55
4	97.68	77.23	81.27	57.69	79.38	44.55
5	97.68	77.23	81.27	57.69	79.38	44.55
6	97.68	77.23	81.27	57.69	79.38	44.55
7	97.68	77.23	81.27	57.69	79.38	44.55

After evaluating the two implementations of the PPM algorithm, with and without neighboring states, it can be observed that the optimal order for both algorithms is 3. The optimal configurations of these two algorithms will now be compared with two prediction methods presented and evaluated in previous works: the Markov model with padding [7], enhanced to use human characteristics, and the LSTM [8]. For the Markov model with padding, the optimal order is 2. The comparisons are presented on both the "Trainees" and "Workers" datasets. Both the capacity to learn and to adapt to new challenges will be measured.

Figure 6 compares the PPM and PPMN methods with the other existing methods in terms of prediction rate. The LSTM network has a very high prediction rate in most of the cases, whereas the PPMN has the top prediction on the "Trainees" dataset with the "100/100" testing method. On new data ("75/25" testing), compared to PPM, the PPMN predicts over 10% more often on the "Workers" dataset and over 20% more often on the "Trainees" dataset, with a prediction rate of 91.18% and 77.23%, respectively.
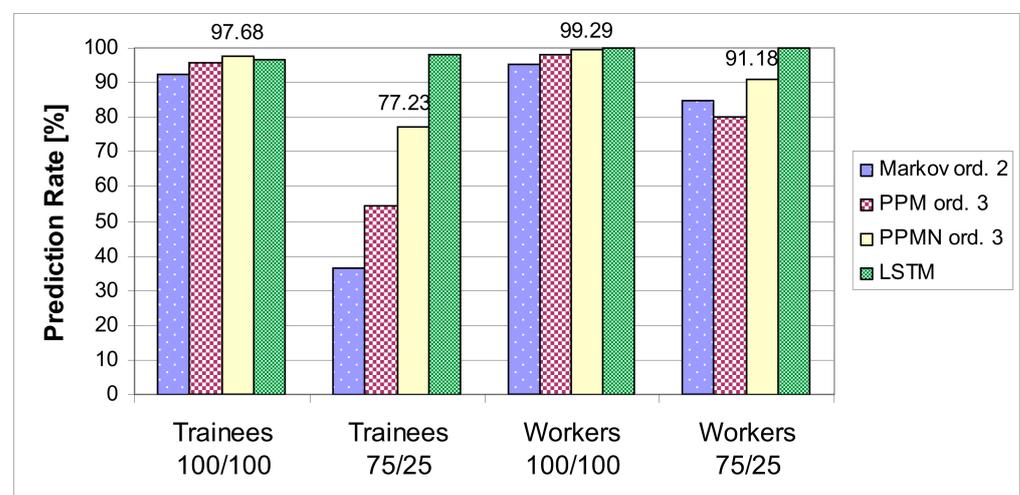

Figure 6. Prediction rate.

Both the PPM and PPMN have a similar prediction accuracy across all datasets, with PPM being slightly higher (see Figure 7). On "100/100 testing", the Markov predictor has the highest prediction accuracy. Although LSTM has the highest prediction rate, it is the one that has the lowest accuracy across all the evaluation scenarios.
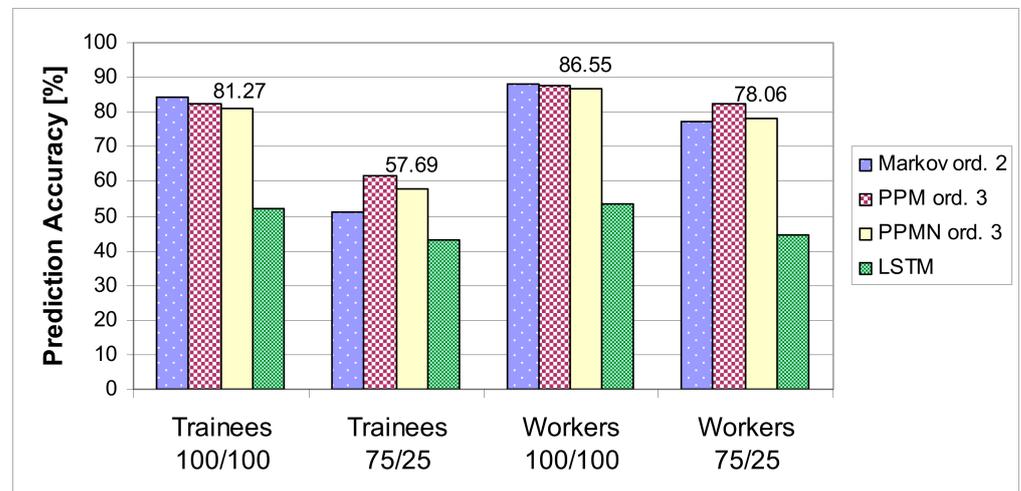

Figure 7. Prediction accuracy.

The coverage measures the capacity of these prediction methods to model existing data and to adapt to new data. As Figure 8 depicts, PPMN is the best prediction method to model existing data and has a coverage of 44.55% on the "Trainees" dataset and 71.18% on the Workers dataset, considering the "75/25" testing method. Taking into account that the coverage is an important efficiency indicator, as it expresses the ratio of correct predictions, these results are remarkable. The combination of different order Markov predictors, as well as the exploration of the neighboring states, proved to be a good solution. The PPMN can easier find matching assembly patterns to provide next step prediction.
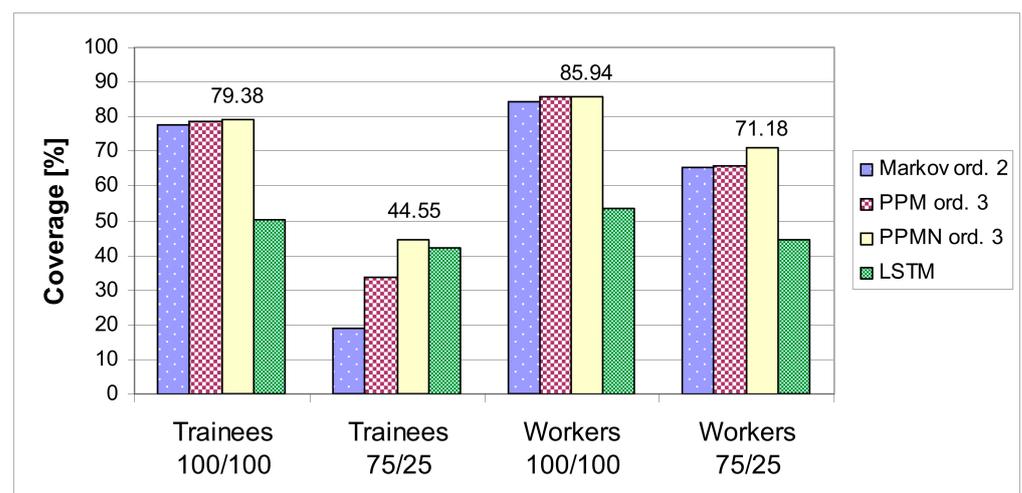

Figure 8. Coverage.

5. Conclusions and Further Work

In this paper a prediction-based assembly support system was presented, which can adaptively direct the workers in their manufacturing activities. The focus is on its prediction module which, in this work, uses the PPM algorithm to provide choices for the next assembly step, which can be helpful especially for inexperienced workers. The evaluations were performed on the assembly data collected within an experiment in

which the participants were 68 trainees and 111 factory workers who had to assemble a customizable modular tablet. The optimal PPM is of order 3. For a higher prediction rate, an enhanced PPM with a neighbor-states checking mechanism (PPMN) was used. Thus, when the algorithm could not find the current state (consisting of the worker's characteristics and the sequence of the last assemblies), it also checked the states which were neighbors from the human characteristics point of view and, in case of success, the next assembly step was determined by majority voting among such existing neighbor states. The PPMN has a significantly higher coverage on new data: 44.55% in the case of trainees and 71.18% in the case of factory workers. It also outperforms the LSTM and the Markov model in terms of coverage.

As concerns directions for further work, the evaluation of Hidden Markov Models and Dynamic Bayesian Network as next assembly step predictors is considered. From a higher perspective, the long-term target for the assembly assistance system is to provide an interactive, tailored experience for trainees and effective training for manual operations. Thus, the prediction system should cover multi-modal interaction for a broad adaptation of instructions from a semantic, detail or type (e.g., visual, audio, text, etc.) perspective. For this, data from many longitudinal studies and experiments on predictor training need to be executed to cover different typologies (e.g., age, gender, handedness, etc.), the basic emotional or mental state of the user during training, previous experience, etc.

Author Contributions: Conceptualization, A.G.; methodology, A.G. and U.F.; software, S.-A.P. and A.G.; validation, S.-A.P.; formal analysis, A.G., S.-A.P. and C.-B.Z.; investigation, A.G., S.-A.P., B.-C.P.; resources, B.-C.P. and A.G.; data curation, S.-A.P. and A.G.; writing—original draft preparation, A.G., S.-A.P., B.-C.P., U.F. and C.-B.Z.; writing—review and editing, A.G., S.-A.P., B.-C.P., C.-B.Z. and U.F.; visualization, S.-A.P. and A.G.; supervision, A.G., C.-B.Z. and F.P.; project administration, B.-C.P. and A.G.; funding acquisition, B.-C.P. and A.G. All authors have read and agreed to the published version of the manuscript.

References

1. Rauch, E.; Linder, C.; Dallasega, P. Anthropocentric perspective of production before and within Industry 4.0. *Comput. Ind. Eng.* **2020**, *139*, 105644. [CrossRef]
2. Romero, D.; Bernus, P.; Noran, O.; Stahre, J.; Fast-Berglund, Å. The operator 4.0: Human cyber-physical systems and adaptive automation towards human-automation symbiosis work systems. In *IFIP International Conference on Advances in Production Management Systems, Proceedings of the APMS 2016: Advances in Production Management Systems. Initiatives for a Sustainable World, Iguassu Falls, Brazil, 3–7 September 2016*; Springer: Berlin/Heidelberg, Germany, 2016; Volume 488, pp. 677–686.
3. Chiacchio, F.; Petropoulos, G.; Pichler, D. *The Impact of Industrial Robots on EU Employment and Wages: A local Labour Market Approach*; Working paper 25186; Bruegel: Saint-Josse-ten-Noode, Belgium, 2018.
4. Funk, M.; Bächler, A.; Bächler, L.; Kosch, T.; Heidenreich, T.; Schmidt, A. Working with augmented reality? A long-term analysis of in-situ instructions at the assembly workplace. In Proceedings of the 10th ACM International Conference on Pervasive Technologies Related to Assistive Environments, Island of Rhodes, Greece, 21–23 June 2017; pp. 222–229.
5. Gellert, A.; Zamfirescu, C.-B. Using Two-Level Context-Based Predictors for Assembly Assistance in Smart Factories. In *Intelligent Methods in Computing, Communications and Control, Proceedings of the 8th International Conference on Computers Communications and Control, Băile Felix, Oradea, Romania, 11–15 May 2020*; Springer: Oradea, Romania, 2020; pp. 167–176.

6.   Gellert, A.; Zamfirescu, C.-B. Assembly support systems with Markov predictors. In Proceedings of the 20th Open Conference of the IFIP WG 8.3 on Decision Support, Wrocław, Poland, 24–26 June 2020. Published in Journal of Decision Systems.

7.   Gellert, A.; Precup, S.-A.; Pirvu, B.-C.; Zamfirescu, C.-B. Prediction-based assembly assistance system. In Proceedings of the 25th International Conference on Emerging Technologies and Factory Automation, Vienna, Austria, 8–11 November 2020; pp. 1065–1068.

8.   Precup, S.-A.; Gellert, A.; Dorobantiu, A.; Zamfirescu, C.-B. Assembly process modeling through long short-term memory. In Proceedings of the 13th Asian Conference on Intelligent Information and Database Systems, Phuket, Thailand, 7–10 April 2021.

9.   Yamada, Y.; Konosu, H.; Morizono, T.; Umetani, Y. Proposal of skill-assist: A system of assisting human workers by reflecting their skills in positioning tasks. In Proceedings of the 1999 IEEE International Conference on Systems, Man, and Cybernetics, Tokyo, Japan, 12–15 October 1999; Volume 4, pp. 11–16.

10.   Tan, Q.; Tong, Y.; Wu, S.; Li, D. Anthropocentric approach for smart assembly: Integration and collaboration. *J. Robot.* 2019. [CrossRef]

11.   Rojas, R.; Rauch, E.; Dallasega, P.; Matt, D.T. Safe human-machine centered design of an assembly station in a learning factory environment. In Proceedings of the International Conference on Industrial Engineering and Operations Management, Bandung, Indonesia, 6–8 March 2018; pp. 403–411.

12.   Elkomy, M.; Abdelrahman, Y.; Funk, M.; Dingler, T.; Schmidt, A.; Abdennadher, S. ABBAS: An adaptive bio-sensors based assistive system. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Osaka, Japan, 8–13 May 2017; pp. 2543–2550.

13.   Funk, M.; Dingler, T.; Cooper, J.; Schmidt, A. Stop helping me—I'm bored! why assembly assistance needs to be adaptive. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and 2015 ACM International Symposium on Wearable Computers, Osaka, Japan, 7–11 September 2015; pp. 1269–1273.

14.   Bertram, P.; Birtel, M.; Quint, F.; Ruskowski, M. Intelligent manual working station through assistive systems. In Proceedings of the 16th IFAC Symposium on Information Control Problems in Manufacturing, Bergamo, Italy, 11–13 June 2018; pp. 170–175.

15.   Mueller, R.; Vette-Steinkamp, M.; Hoerauf, L.; Speicher, C.; Bashir, A. Intelligent and flexible worker assistance systems. In Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2018), Funchal, Madeira, Portugal, 27–29 January 2018; Volume 2: HUCAPP, pp. 77–85.

16.   Gorecky, D.; Khamis, M.; Mura, K. Introduction and establishment of virtual training in the factory of the future. *Int. J. Comput. Integr. Manuf.* **2017**, *30*, 182–190. [CrossRef]

17.   Tocu, N.-A.; Gellert, A.; Stefan, I.-R.; Nitescu, T.-M.; Luca, G.-A. The impact of virtual reality simulators in manufacturing industry. In Proceedings of the 12th Annual International Conference on Education and New Learning Technologies, Palma de Mallorca, Spain, 6–8 July 2020.

18.   Korn, O.; Schmidt, A.; Hörz, T. Assistive systems in production environments: Exploring motion recognition and gamification. In Proceeding of the 5th International Conference on Pervasive Technologies Related to Assistive Environments, Heraklion, Crete, Greece, 9–13 June 2012. [CrossRef]

19.   Longo, F.; Nicoletti, L.; Padovano, A. Smart operators in industry 4.0: A human-centered approach to enhance operators' capabilities and competencies within the new smart factory context. *Comput. Ind. Eng.* **2017**, *113*, 144–159. [CrossRef]

20.   Lai, Z.-H.; Tao, W.; Leu, M.C.; Yin, Z. Smart augmented reality instructional system for mechanical assembly towards worker-centered intelligent manufacturing. *J. Manuf. Syst.* **2020**, *55*, 69–81. [CrossRef]

21.   Loskyll, M.; Heck, I.; Schlick, J.; Schwarz, M. Context-based orchestration for control of resource-efficient manufacturing processes. *Future Internet* **2012**, *4*, 737–761. [CrossRef]

22.   Gellert, A.; Florea, A. Web prefetching through efficient prediction by partial matching. *World Wide Web J.* **2016**, *19*, 921–932. [CrossRef]

23.   Gellert, A.; Florea, A.; Vintan, M.; Egan, C.; Vintan, L. Unbiased Branches: An Open Problem. In Proceeding of the 12th Asia-Pacific Computer Systems Architecture Conference (ACSAC 2007), Seoul, Korea, 23–25 August 2007; pp. 16–27.