CoBaIR: A Python Library for Context-Based Intention Recognition in Human-Robot-Interaction

Adrian Lubitz[1], Lisa Gutzeit[1] and Frank Kirchner[1,2]

***Abstract*— Human-Robot Interaction (HRI) becomes more and more important in a world where robots integrate fast in all aspects of our lives but HRI applications depend massively on the utilized robotic system as well as the deployment environment and cultural differences. Because of these variable dependencies it is often not feasible to use a data-driven approach to train a model for human intent recognition. Expert systems have been proven to close this gap very efficiently. Furthermore, it is important to support understandability in HRI systems to establish trust in the system. To address the above-mentioned challenges in HRI we present an adaptable python library in which current state-of-the-art Models for context recognition can be integrated. For Context-Based Intention Recognition a two-layer Bayesian Network (BN) is used. The bayesian approach offers explainability and clarity in the creation of scenarios and is easily extendable with more modalities. Additionally, it can be used as an expert system if no data is available but can as well be fine-tuned when data becomes available.**

I. INTRODUCTION

Our day-to-day lives are becoming increasingly involved with robotic devices. The industry is currently changing from static robotic environments to dynamic environments where humans collaborate with robots instead of operating robots. In private homes, robotic applications like robot vacuums and digital assistants for home automation are becoming regular household items [1]. Although this movement is drastically changing our society, interactions between humans and robots are very command-driven and unnatural in contrast to Human-Human Interaction (HHI) [2], [3]. While Large Language Models (LLM) like *GPT-3*[4], *BERT*[5], *LLaMA*[6], and *LaMDA*[7] show very promising results in general language understanding, they have problems with biases, alignment, uncertainty estimation, and most importantly they lack multimodal understanding of their surroundings which is a key feature for Intention Recognition needed in modern HRI applications [8]. In general, data-driven Intention Recognition systems [9], [10] have the drawback of relying on large and complex (high dimensional) human-robot interaction data. Because in most scenarios data is not available or impractical to record, data-driven approaches are not suitable for this problem. Bayesian context-based Intention Recognition is an approach to overcome those limitations and offer an expert system that can be fine-tuned with data when it becomes available. Existing research in this direction [11], [12], [13], [14], [15], [16] is promising but introduces very complex network structures which induce the need for the designer of an HRI scenario to have a profound knowledge of Bayesian probability theory. Furthermore, it makes adaptations and re-configuration of the network cumbersome. To overcome the aforementioned limitations of current systems we propose a two-layer BN for context-based Intention Recognition. The simple structure enables us to make several optimizations that allow the designer to concentrate on the HRI scenario instead of Bayesian probability theory.

In [17] we proposed a first concept on how the design for context-based Bayesian Intention Recognition in HRI scenarios can be described in a more compact and intuitive way. In this paper, we introduce **Co**ntext-**Ba**sed **I**ntention **R**ecognition (*CoBaIR*), a python software library that comes with the power to infer intentions from the current context — context describes every observable aspect in an HRI scenario. Furthermore, *CoBaIR* pays great attention to the design process of HRI scenarios. It provides a configuration format that decreases the number of values that need to be set during the design process of the Bayesian Network from an exponential to a linear scale. Additionally, it provides a Graphical User Interface (GUI) which visualizes the two-layer BN with its weights and offers an intuitive way of configuring it.

This paper starts by pointing out the key challenges in Intention Recognition for HRI in Section II-A. In Section III we provide a detailed view of the proposed two-layer BN structure and highlight how this structure can make the process of designing HRI scenarios easier and faster. In Section IV we highlight the advantages of the proposed approach. We point out the most important features of the python implementation in Section V. In Section VI we show an example of how the described python library was used in a research project. Finally, in Section VII we conclude with the interpretation of the results and an outlook on future work.

II. RELATED WORK

During the extensive literature review conducted for this paper, it was observed that existing intention recognition implementations were often tightly coupled with specific modalities [18], [19] or designed exclusively for particular scenarios [20], [21]. In some cases, these limitations were

*This work was supported through a grant of the German Federal Ministry of Economic Affairs and Climate Action (BMWi, FKZ 50 RA 2022)

[1] Robotics Research Group, Department of Computer Science, University of Bremen, 28359 Bremen, Germany alubitz@uni-bremen.de, lisa.gutzeit@uni-bremen.de, kirchner@informatik.uni-bremen.de

[2] German Research Center for Artificial Intelligence, 28359 Bremen, Germany frank.kirchner@dfki.de

found to coexist [22], further hindering the applicability of such implementations. However, as part of our objective to provide a generic framework for HRI within the KiMMI Project, we recognized the need for a solution that could exhibit high flexibility and adaptability across various modalities and scenarios.

A. Challenges in Intention Recognition

HRI depends in many ways on the scenario at hand. For Intention Recognition we define the following challenges that need to be addressed in order to implement natural and meaningful HRI systems:

Hardware constraints: Just like humans, robots come in all shapes and colors. More precisely, social robots for HRI have different sensing modalities to perceive the human and its surrounding. While some robots are equipped with stereo-vision or RGBD-camera systems and multiple microphones for echolocation, as well as LiDAR for navigation, a simple digital assistant may only be equipped with one microphone. The designer of the HRI scenario needs to know which sensory modalities are available for the system in question. [23], [24], [25]

Application specifics: Furthermore, the designer needs to know about the application scenario which can vary from space applications over industrial to domestic applications. In all of these different application scenarios gestures, voice commands, etc. can mean different things. [23], [25]

Cultural differences: Research on cultural differences in social robotics is an often neglected topic although social robots will be deployed in multi-cultural place e.g. airports in the future in a more and more globalized world. One behavior may have different meanings in different cultures. Therefore the designer must be aware of the cultural differences, and for different cultures, different HRI scenarios must be designed. [25], [26]

Individual differences: When we think about developing robots that interact with humans in a very intuitive way we need to ask ourselves what is intuitive for us. Intuitive may be slightly different from person to person even within one cultural group. Humans interact slightly different on an individual level based on their knowledge about the interaction partner. If the interaction partner is not known a default is chosen which allows for adaptation in the future. While this behavior is very subtle and unconscious in HHI it is an important factor while designing HRI scenarios with the possibility for inter-personal adaptation. [24], [25], [26]

Trust & Acceptance: Trust in a robotic system is less scenario specific than the aforementioned challenges and can therefore not as obviously be integrated into the design process of an HRI scenario. Trust is primarily connected with the human's expectation of the behavior of a robot. If the robot behaves accordingly to the human's expectations the human can foresee the behavior and build a model of trust for the robot's abilities. Secondarily, it is connected with the explainability of a behavior. If the human is not able to foresee the robot's behavior because it is, e.g. not completely deterministic or too complex to foresee, the human will seek reasons for the observed behavior. If reasons can be found trust can still be maintained, while if no reason can be found the trust will decrease immensely. This model of trust can be very complex in a way that trust exists for specific abilities but not for others. The trust for specific abilities weighted with their specific importance for the human determines the acceptance in the system. [24], [23], [25]

In Section III - VI we illustrate how we address these challenges, which advantages arise from the proposed approach, how it is implemented, and how it can be used to model an HRI scenario.

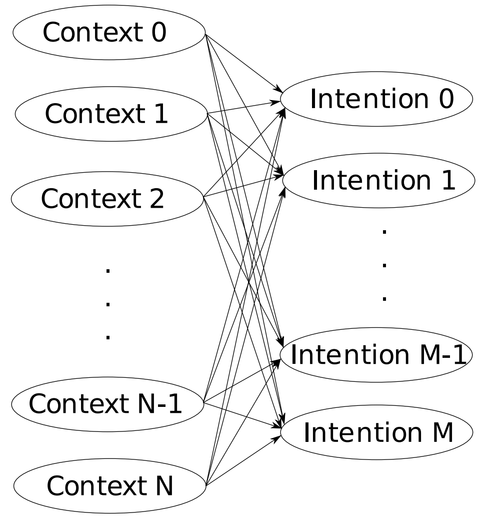

Fig. 1. The architecture of the two-layer Bayesian network allows for a high degree of flexibility

III. Architecture of the Two-Layer Bayesian Network for Context-Based Intention Recognition

In Section II-A we highlighted some of the key challenges in Intention Recognition in HRI. We propose a two-layer BN to address these challenges in a computational and data efficient as well as an intuitive way. The general structure of the BN for context-based Intention Recognition is depicted in Figure 1. In [12] and [15] Bayesian Networks (BNs) with 3 or more layers are proposed to additionally model actions. We believe the two-layered structure comes with several advantages over using three or more layers. From a usability point of view, it allows us to assume that the designer of the HRI scenario has no prior knowledge about Bayesian probability in general and BNs in specific.

In this way every contexts can be treated in the same way, as an observable phenomenon. Using further layers would give actions, as suggested by [12], a special meaning.

This special meaning however is not always valid and furthermore, it introduces a bias towards actions in the design of HRI scenarios. In cases where external context is of more importance than the performed action, the introduced bias towards actions can distort the design of the scenario.

In cases where external factors like the time have an influence on the intention the action itself should not play a predominant role. The action of grasping a mug could lead to the intention `make coffee` but at night the intention becomes more unrealistic and a higher probability should be given to the intention `store mug`. This is a very simple example of how the action bias could lead to the intention `make coffee` at night where the correct intention should be `store mug`.

Modeling relevant observable phenomena as context reduces the action bias and the complexity of the BN and allows the designer of the scenario to concentrate on the specifics of the scenario without the need for in-depth knowledge about the underlying probabilistic modeling.

The final step we took to uncouple probabilistic modeling from the intuitive design of HRI scenarios is to make some assumptions over the given two-layer structure. These assumptions reduce the probabilistic notion and drastically reduce the number of values that need to be set by a human expert to describe a scenario.

Independent context and intentions

The basic assumption is that all intentions are independent of each other as well as all contexts are independent of each other. This allows for the strict two-layer structure which has only connections between context and intentions as depicted in Figure 1.

Binary intentions

We consider all intentions as binary — an intention is either present or not. This allows us to concentrate on the positive (an intention is present) case while designing the scenario and consider the negative (an intention is not present) case as its complement while calculating the probabilistic model. This small constraint already cuts the number of values that need to be set by the human expert in half. Contexts on the other hand can have as many discrete instantiations as needed to create a meaningful HRI scenario.

Single Condition Assumption

We make the Single Condition Assumption (SCA) which implies that every context has an individual and independent influence on a specific intention. Using this assumption, we can approximate the conditional probability $P(I_m|C_{1,l}, C_{2,l}, ..., C_{k,l})$ as the average over all single condition probabilities $P(I_m|C_{k,l})$, where I_m is the *m*-th intention and $C_{k,l}$ is the *l*-th instantiation of the *k*-th context. This allows us to calculate the conditional probability of the *m*-th intention given the first instantiation for all contexts in the following way:

$$P(I_m|C_{1,l}, C_{2,l}, ..., C_{k,l}) = \frac{\sum_{\hat{k}=1}^{k} P(I_m|C_{\hat{k},l})}{k} \qquad (1)$$

Influence values on a Likert scale

We generalize the single condition probability $P(I_m|C_{k,l})$ as an influence value $v_{k,l,m}$ on a six-point Likert scale [27] for every context-intention-tuple $(C_{k,l}, I_m)$ to make them more manageable. The scale is mapped in the following fashion: $0 \mapsto 0\%; 1 \mapsto 5\%; 2 \mapsto 25\%; 3 \mapsto 50\%; 4 \mapsto 75\%; 5 \mapsto 95\%$

The above-mentioned assumptions reduce the amount of values that need to be set by a human expert from an exponential growth given through

$$V(i, j, c, n) = \sum_{\hat{j}=1}^{j} c_{\hat{j}} + i \times \prod_{\hat{i}=1}^{i} n_{\hat{i}} \times \prod_{\hat{j}=1}^{j} c_{\hat{j}} \qquad (2)$$

to a linear growth given through

$$V(i, j, c) = (i + 1) \times \sum_{\hat{j}=1}^{j} c_{\hat{j}} \qquad (3)$$

where V is the number of values to be set, i is the number of intentions, j is the number of contexts, c_j is the number of context instantiations for the *j*-th context, n_i is the number of intention instantiations for the *i*-th intention. The $\sum^{j} c_j$ in Equations 2 and 3 describes the amount of *a priori* probabilities for all context instantiations. The remaining term describes the values needed to fill the Conditional Probability Tables (CPTs) manually for Equation 2 and automatically for Equation 3. The SCA contributes in a huge way to the ease of designing HRI scenarios but ignores cases in which joint probabilities are necessary. An example could be the handling of voice commands through a robot that is able to estimate directed speech over Visual Voice Activity Detection (VVAD) as shown in [28]. The robot should infer the intention `pick up tool` with a higher probability if the speech command for picking up a tool was emitted **AND** the speech was directed towards the robot than one of them individually. Only the speech command should have a high probability to infer `pick up tool` but there is still a chance that the robot was picking up noise or the speech was not directed towards the robot. The context of directed speech on the other hand does not have a high probability for any intention individually. For those special cases, we provide the possibility to set (partially) conditioned influence values containing multiple contexts that provide more information when combined. [17]

While the two-layer BN was originally not designed to handle temporal dependencies, it is possible to model the previously inferred intention as context. Using this recursive pattern, it is possible to model a temporal dependency under the Markov assumption [29]. This allows to model situations where it becomes more or less likely to infer a specific intention if the same or another intention was inferred before. In a conversational setting it would be possible to adjust how shy a robot is. Given that the context `eye contact` is `true` the robot infers the intention `human interested in conversation` and could initiate a conversation. If the robot is a little shy `last intention` could be modeled as context and the intention recognition could be configured in a way that `eye contact` must be `true`, `human interested in conversation` must be `true` to infer the intention `human wants robot to initiate conversation`. This shifts the initiation from the robot side by one timestep.

IV. Advantages of a two-layer Bayesian Network for Context-Based Intention Recognition

CoBaIR uses a two-layer BN to represent the dependencies between contexts and intentions. In this section we want to highlight the key advantages of this structure:

Flexibility

The biggest advantage of the architecture is its flexibility. It allows for the usage of any algorithm for context creation, whether it be probabilistic, heuristic, data-driven or any other approach. The generated contexts will be used as the input to the two-layer BN which fuses the context information to jointly infer a probability distribution over all possible intentions. On the one hand, using *CoBaIR* as an expert system the simplifications explained in Section III allow the human designer to create a scenario in a fast and intuitive manner. Fast design and adaptation of HRI scenarios helps researchers to concentrate on the specifics of an experiment and therefore reach results faster and more reliably without any deeper knowledge about Bayesian probability and how to configure BNs. On the other hand the two-layer BN can be trained or fine-tuned with data, which allows to gradually shift from a system trained by an expert to a data-driven approach. Furthermore, the simple structure of the BN allows for the easy removal and addition of contexts. This makes an iterative prototyping approach, where the HRI scenario is build up over time, possible.

Uncertainty Quantification and Explainability

Uncertainty Quantification (UQ) is an often neglected topic, especially in the field vision based tasks [30]. The Bayesian approach for Intention Recognition offers the implicit advantage that it comes with a inherently good UQ due to the probabilistic nature of the model. Additionally, the compact structure of the two-layer BN allows users to easily identify the contexts that played the predominant role in the decision making. Using this interpretable compact structure we are able to generate explanations to understand the decisions made by a robot using *CoBaIR* as its Intention Recognition system. Explainability and UQ in a system strongly increases the trust in the system and therefore the acceptance to use the system in general [31].

Modularity

Another advantage of the described architecture is its modularity. Using a two-layer BN to fuse the output of different modules that provide contexts makes it possible to use existing solutions for context creation, like PAZ [32] which provides a large variety of models for visual perception, as well as use case specific models that need to be trained from scratch. Furthermore, it is possible to switch seamlessly between different models on the fly for evaluation and optimization of HRI scenarios.

Handling missing input

While most data-driven approaches have problems handling missing input [33] and additional data needs to be recorded or artificially generated, BNs provide the advantage of defining *a priori* probabilities for the input. The *a priori* probabilities for the context can be estimated by an expert or calculated from a few observations. In this way knowledge about missing input can be incorporated and during inference time the missing inputs will handled accordingly.

The above-mentioned advantages of the two-layer BN highlight why we think a two-layer BN is suitable to enhance the design, inference and explainability of Intention Recognition in HRI scenarios.

V. CoBaIR: a Python Library

CoBaIR is a python library for **Co**ntext-**Ba**sed **I**ntention **R**ecognition. The library allows to create complex HRI scenarios in a fast and intuitive manner. Furthermore, it provides a GUI which visualizes the underlying two-layer BN and guides through the configuration procedure.

CoBaIR is divided into two parts:

A. *Core library*

The core library provides all the key features described in Section III to make the design of HRI scenarios intuitive and fast. It mainly provides the class `BayesNet` which handles the creation of the two-layer BN from a given configuration. The configuration is provided in YAML and its fields and format is depicted in Listing 1. The format contains the fields *contexts*, *instantiations*, *intentions* and *decision_threshold*. *contexts* gives names to the observable phenomena, like *weather*. *instantiations* are the discrete instantiations of that phenomenon, like *cloudy, rainy, sunny*. The binary *intentions* are the inferable intentions in the scenario, like *turn on sprinkler*. All *instantiations* have an a priori probability which needs to be set and furthermore the *instantiations* have an influence value for each *intention*. Additionally, there is a field *decision_threshold* which can be a value between 0 and 1 and denotes the threshold an intention's likelihood needs to surpass during inference to be considered as the inferred intention. If the likelihood of the most likely intention is below the threshold, `None` is returned as the inferred intention.

Listing 1. Configuration format for Intention Recognition with *CoBaIR*

```
contexts:
  context 1:
    instantiation 1 : float
    .
    instantiation m_1 : float
  context n:
    instantiation 1 : float
    .
    instantiation m_n : float
```

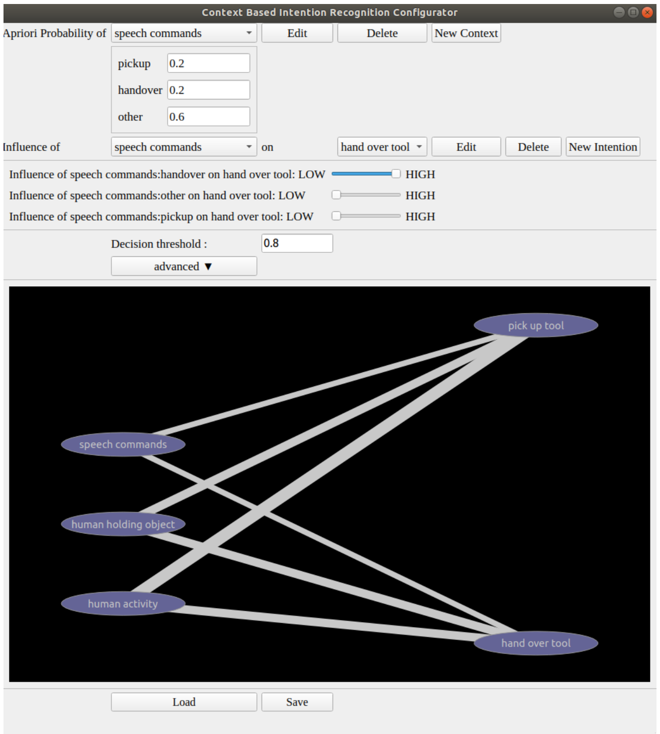

Fig. 2. The GUI of *CoBaIR* supports the designer of an HRI scenario.

```
intentions:
  intention 1:
    context 1:
        instantiation 1: int # one out of [5, 4, 3, 2, 1, 0]
        .
        instantiation m_1: int # one out of [5, 4, 3, 2, 1, 0]
    context n:
        instantiation 1: int # one out of [5, 4, 3, 2, 1, 0]
        .
        instantiation m_n: int # one out of [5, 4, 3, 2, 1, 0]
  intention p:
    context 1:
        instantiation 1: int # one out of [5, 4, 3, 2, 1, 0]
        .
        instantiation m_1: int # one out of [5, 4, 3, 2, 1, 0]
    context n:
        instantiation 1: int # one out of [5, 4, 3, 2, 1, 0]
        .
        instantiation m_n: int # one out of [5, 4, 3, 2, 1, 0]
decision_threshold: float
```

The core library provides the means to validate, load and save the configuration. With the information from the configuration a fully defined two-layer BN will be created. bnlearn [34] is utilized as a backend to handle the two-layer BNs and do inference on them. The Application Programming Interface (API) is fully documented an publicly available under https://dfki-ric.github.io/CoBaIR/API.

Graphical User Interface

On top of the optimizations we described in Section III, we provide researchers and practitioners with a helpful GUI depicted in Figure 2. The GUI supports the designer to create a configuration and makes sure it is always valid and provides helpful insights in the case of an invalid configuration. Furthermore, it visualizes the two-layer BN in a live view which helps to keep a good overview of the configuration.
The complete Python software package is available on PyPI. Additionally, the code is open sourced on GitHub and open for contributions.

VI. Application of CoBaIR

We applied *CoBaIR* in an interaction scenario in the project KiMMI SF[1] to infer the intentions of a human operator. This user controlled a simulated human during the interaction with the simulated robotic arm from Universal Robot UR5 mounted on a movable base. The simulated environment is depicted in Figure 3. The human can be navigated through the environment, which contains shelves in which different tools are stored, a working station with an inactive robotic system MANTIS[2] that should be repaired, and on the left back side a dark area. Based on the human behavior the robotic system should infer the human's intention in order to react accordingly. For example, if the human wants to repair the inactive MANTIS robot, the supporting robot should bring the tool which is stored in the shelf, which is shown on the right side in Figure 3.

The BN used to infer the human intentions is shown in Figure 4. The human behavior is measured based on four contexts: *hand opening*, *human pose*, *location of interest*, and *speech commands*, where the first three are determined in the simulation and the speech commands are captured from the operator of the simulation with a microphone. Each context is discretized to different values, e.g., *hand opening* can be either *open* or *closed*. This is shown on the left side of Figure 4 for each context. In the presented scenario, based on the given contexts, the following five intentions should be inferred: 1. *go work station*, i.e., the human wants to go to the work station; 2. *go dark space*, i.e., the human want to go to the dark area; 3. *robot bring tool*, i.e., the human wants the robot to bring the tool stored in the shelf; 4. *robot stop*, i.e., the human wants the robot to stop its current action; 5. *robot store tool*, the human wants the robot to store the tool back to the shelf.

Using *CoBaIR* and the included GUI shown in Figure 2, the BN shown in Figure 4 could easily be designed and optimized for the described scenario. With the resulting BN all human intentions could be reliably inferred and the correct reactions of the UR5 could be triggered to realize a successful interaction between the human and the robotic system.

During the KiMMI SF project we followed an agile development process which led to multiple incremental as well as complete changes in the design of the HRI scenario. *CoBaIR* enabled us to incorporate these changes fast and effectively in the development process.

VII. Conclusion and Future Work

We presented the python library *CoBaIR* in this paper. We demonstrated that the concept from [17] using a two-layer BN with the assumptions highlighted in III can be effectively implemented. Additionally, we provided a GUI to visualize and guide the design process for HRI scenarios. In Section VI we showed that *CoBaIR* was successfully used

[1] https://robotik.dfki-bremen.de/en/research/projects/kimmi-sf/
[2] https://robotik.dfki-bremen.de/en/research/robot-systems/mantis/

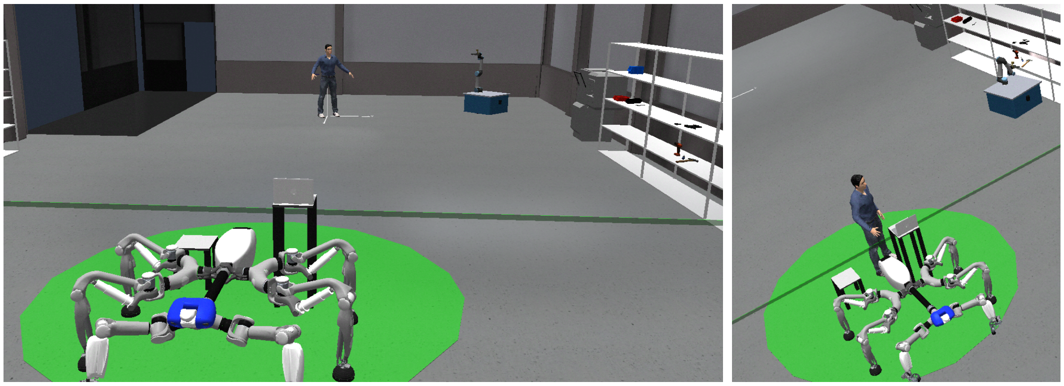

Fig. 3. Screenshots of the HRI application scenario, in which a simulated human interacts with the UR5 robot mounted on a movable base. The scenario includes a shelf with tools stored in it, a dark area which should be lighted by the UR5 if the human needs support in the dark area, and a work station highlighted in green area where the human can work on the robotic system MANTIS and the UR5 should assist by bringing the desired tool.

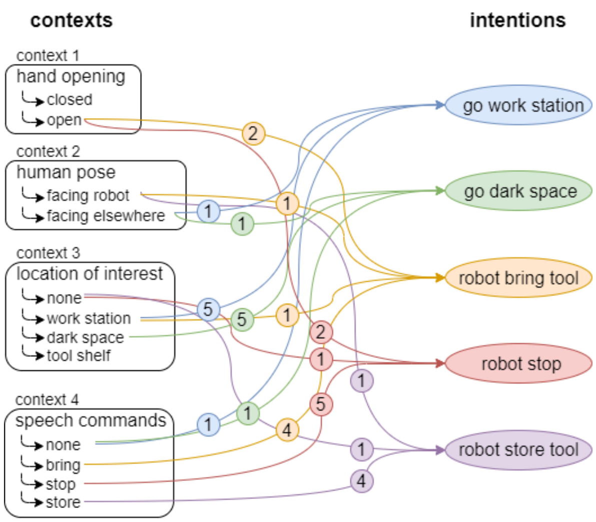

Fig. 4. BN of the application scenario. Based on four different contexts (left side), five possible human intentions (right side) should be inferred. Arrows indicate the influence of the possible values of the contexts with small numbers representing the assigned influence value.

in the KiMMI SF project. We advise practitioners to use the state-of-the-art models for context recognition from the open source library PAZ [32] which is constantly updated with perception models for autonomous systems. In the future we plan on providing tutorials and examples on how to use *CoBaIR* with PAZ. While it is theoretically possible to train or fine-tune the two-layer BN with data from the scenario,so far the fine-tuning and data-driven training from scratch was not tested. In future works we want to investigate the capability of data-driven training and fine-tuning within *CoBaIR*. Additionally, we want to quantify the effect in terms of quality and speed which *CoBaIR* has on the design of HRI scenarios, conducting user studies comparing the time and cognitive load to create a HRI scenario using our solution in contrast to creating a HRI scenario using CPTs for the creation of the BN.. We will be incorporating *CoBaIR* in future projects and by making *CoBaIR* open source we hope to provide a helpful tool for researchers and practitioners in HRI all over the world. Everyone is welcome to use, give feedback and contribute to *CoBaIR* through GitHub.

ACKNOWLEDGMENT

This work was supported through a grant of the German Federal Ministry of Economic Affairs and Climate Action (BMWi, FKZ 50 RA 2022).

REFERENCES

[1] L. Fortunati, A. Esposito, and G. Lugano, "Introduction to the Special Issue "Beyond Industrial Robotics: Social Robots Entering Public and Domestic Spheres"," *The Information Society*, vol. 31, no. 3, pp. 229–236, May 2015.

[2] H. Yan, M. H. Ang, and A. N. Poo, "A Survey on Perception Methods for Human–Robot Interaction in Social Robots," *International Journal of Social Robotics*, vol. 6, no. 1, pp. 85–119, Jan. 2014.

[3] D. Y. Y. Sim and C. K. Loo, "Extensive assessment and evaluation methodologies on assistive social robots for modelling human–robot interaction – A review," *Information Sciences*, vol. 301, pp. 305–344, Apr. 2015.

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language Models are Few-Shot Learners," July 2020, arXiv:2005.14165 [cs]. [Online]. Available: http://arxiv.org/abs/2005.14165

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," May 2019, arXiv:1810.04805 [cs]. [Online]. Available: http://arxiv.org/abs/1810.04805

[6] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "LLaMA: Open and Efficient Foundation Language Models," Feb. 2023, arXiv:2302.13971 [cs]. [Online]. Available: http://arxiv.org/abs/2302.13971

[7] R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H.-T. Cheng, A. Jin, T. Bos, L. Baker, Y. Du, Y. Li, H. Lee, H. S. Zheng, A. Ghafouri, M. Menegali, Y. Huang, M. Krikun, D. Lepikhin, J. Qin, D. Chen, Y. Xu, Z. Chen, A. Roberts, M. Bosma, V. Zhao, Y. Zhou, C.-C. Chang, I. Krivokon, W. Rusch, M. Pickett, P. Srinivasan, L. Man, K. Meier-Hellstern, M. R. Morris, T. Doshi, R. D. Santos, T. Duke, J. Soraker, B. Zevenbergen, V. Prabhakaran, M. Diaz, B. Hutchinson, K. Olson, A. Molina, E. Hoffman-John, J. Lee, L. Aroyo, R. Rajakumar, A. Butryna, M. Lamm, V. Kuzmina, J. Fenton, A. Cohen, R. Bernstein, R. Kurzweil, B. Aguera-Arcas, C. Cui, M. Croak, E. Chi, and Q. Le, "LaMDA: Language Models for Dialog Applications," Feb. 2022, arXiv:2201.08239 [cs]. [Online]. Available: http://arxiv.org/abs/2201.08239

[8] A. Tamkin, M. Brundage, J. Clark, and D. Ganguli, "Understanding the Capabilities, Limitations, and Societal Impact of Large Language Models," Feb. 2021.

[9] Z. Wang, B. Wang, H. Liu, and Z. Kong, "Recurrent convolutional networks based intention recognition for human-robot collaboration tasks," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2017, pp. 1675–1680.

[10] Z. Liu and J. Hao, "Intention Recognition in Physical Human-Robot Interaction Based on Radial Basis Function Neural Network," *Journal of Robotics*, vol. 2019, p. e4141269, Apr. 2019.

[11] L. Pereira and T. A. Han, *Intention Recognition via Causal Bayes Networks Plus Plan Generation*, Oct. 2009, vol. 5816.

[12] L. M. Pereira and H. T. Anh, "Intention Recognition with Evolution Prospection and Causal Bayes Networks," in *Computational Intelligence for Engineering Systems: Emergent Applications*, ser. Intelligent Systems, Control and Automation: Science and Engineering, A. Madureira, J. Ferreira, and Z. Vale, Eds. Dordrecht: Springer Netherlands, 2011, pp. 1–33.

[13] D. Panagopoulos, G. Petousakis, R. Stolkin, G. Nikolaou, and M. Chiou, "A Bayesian-Based Approach to Human Operator Intent Recognition in Remote Mobile Robot Navigation," Sept. 2021.

[14] R. Kelley, A. Tavakkoli, C. King, A. Ambardekar, M. Nicolescu, and M. Nicolescu, "Context-Based Bayesian Intent Recognition," *IEEE Transactions on Autonomous Mental Development*, vol. 4, no. 3, pp. 215–225, Sept. 2012.

[15] K. A. Tahboub, "Intelligent Human-Machine Interaction Based on Dynamic Bayesian Networks Probabilistic Intention Recognition," *Journal of Intelligent and Robotic Systems*, vol. 45, no. 1, pp. 31–52, Jan. 2006.

[16] S. Jain and B. Argall, "Recursive Bayesian Human Intent Recognition in Shared-Control Robotics," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2018, pp. 3905–3912.

[17] A. Lubitz, O. Arriaga, T. Hassan, N. Hoyer, and E. A. Kirchner, "A Bayesian Approach to Context-based Recognition of Human Intention for Context-Adaptive Robot Assistance in Space Missions," 2022.

[18] Y.-M. Jang, R. Mallipeddi, S. Lee, H.-W. Kwak, and M. Lee, "Human intention recognition based on eyeball movement pattern and pupil size variation," *Neurocomputing*, vol. 128, pp. 421–432, Mar. 2014.

[19] D. Zhang, L. Yao, K. Chen, S. Wang, X. Chang, and Y. Liu, "Making Sense of Spatio-Temporal Preserving Representations for EEG-Based Human Intention Recognition," *IEEE Transactions on Cybernetics*, vol. 50, no. 7, pp. 3033–3044, July 2020.

[20] Q. Wang, W. Jiao, R. Yu, M. T. Johnson, and Y. Zhang, "Virtual Reality Robot-Assisted Welding Based on Human Intention Recognition," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 2, pp. 799–808, Apr. 2020.

[21] Y. Xing, C. Lv, H. Wang, H. Wang, Y. Ai, D. Cao, E. Velenis, and F.-Y. Wang, "Driver Lane Change Intention Inference for Intelligent Vehicles: Framework, Survey, and Challenges," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4377–4390, May 2019.

[22] C. Zhu, W. Sun, and W. Sheng, "Wearable sensors based human intention recognition in smart assisted living systems," in *2008 International Conference on Information and Automation*, June 2008, pp. 954–959.

[23] T. B. Sheridan, "Human–Robot Interaction: Status and Challenges," *Human Factors*, vol. 58, no. 4, pp. 525–532, June 2016.

[24] B. Hayes and B. Scassellati, "Challenges in Shared-Environment Human-Robot Collaboration," 2007.

[25] C. Jost, B. L. Pévédic, T. Belpaeme, C. Bethel, D. Chrysostomou, N. Crook, M. Grandgeorge, and N. Mirnig, *Human-Robot Interaction: Evaluation Methods and Their Standardization*. Springer Nature, May 2020.

[26] C. Nehaniv, K. Dautenhahn, J. Kubacki, M. Haegele, C. Parlitz, and R. Alami, "A methodological approach relating the classification of gesture to identification of human intent in the context of human-robot interaction," in *ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication, 2005.*, Aug. 2005, pp. 371–377.

[27] R. Likert, "A technique for the measurement of attitudes," *Archives of Psychology*, vol. 22 140, pp. 55–55, 1932.

[28] A. Lubitz, M. Valdenegro-Toro, and F. Kirchner, "The VVAD-LRS3 Dataset for Visual Voice Activity Detection," in *7th International Conference on Human Computer Interaction Theory and Applications*, Mar. 2023, pp. 39–46.

[29] A. A. Markov, "The Theory of Algorithms," *Journal of Symbolic Logic*, vol. 18, no. 4, pp. 340–341, 1953, publisher: Association for Symbolic Logic.

[30] M. Valdenegro-Toro, "I Find Your Lack of Uncertainty in Computer Vision Disturbing," Apr. 2021.

[31] L. Sanneman and J. A. Shah, "Trust Considerations for Explainable Robots: A Human Factors Perspective," May 2020.

[32] O. Arriaga, M. Valdenegro-Toro, M. Muthuraja, S. Devaramani, and F. Kirchner, "Perception for Autonomous Systems (PAZ)," Oct. 2020.

[33] N. Ipsen, P.-A. Mattei, and J. Frellsen, "How to deal with missing data in supervised deep learning?" p. 6, 2020.

[34] E. Taskesen, "Learning Bayesian Networks with the bnlearn Python Package." 1 2020. [Online]. Available: https://erdogant.github.io/bnlearn