Exploring the Reciprocal Influence of Artificial Intelligence and End-User Development

Barbara Rita Barricelli[1], Daniela Fogli[1]

[1]Department of Information Engineering, University of Brescia, Via Branze 38, Brescia, Italy

Abstract

This paper explores the reciprocal influence between Artificial Intelligence (AI) features of modern systems and End-User Development (EUD) activities aimed at adapting systems' behavior to users' needs and preferences. To improve the quality of life of people who are called on to use AI-infused systems and customize them, new methods and techniques for EUD should be studied. EUD could be of help in exploiting AI algorithms to collect information about users and to offer them advanced interaction modalities. The paper explores these possibilities through the analysis of two application domains where the effective combination of AI and EUD might play a crucial role in the future.

Keywords

artificial intelligence, collaborative robot, end-user development, smart environment, virtual assistant

1. Introduction

The first approaches to End-User Development (EUD) date back to the beginning of this millennium. Thanks to the European Network of Excellence (EUD-Net) established in 2003, this research area rapidly developed. In 2006, EUD-Net curated a book describing the earlier work on EUD [1] about approaches to the creation of EUD environments tailored to domain experts (e.g., [2, 3, 4]), methods and techniques to support programming by end users (e.g., [5, 6]), and conceptual frameworks for end-user development like meta-design [7] and semiotic engineering [8]. The book also reported a first definition of EUD, coined within the EUD-network. EUD was defined as "a set of methods, techniques, and tools that allow users of software systems, who are acting as non-professional software developers, at some point to create, modify, or extend a software artifact" [9]. That definition reflects the time in which it was conceived, in fact it focused on software artifacts only. Recently, advancements in the EUD field have been published in a new Springer book [10], which considers further application domains of EUD, such as Internet of Things (IoT), big data and virtual reality. Particularly, the need of customizing and shaping the behavior of digital artifacts encompassing hardware technology and IoT has become more and more urgent. Specifically, EUD approaches have been proposed to create and modify smart environments by their own inhabitants [11, 12, 13, 14].

Proceedings of CoPDA2022 - Sixth International Workshop on Cultures of Participation in the Digital Age: AI for Humans or Humans for AI? June 7, 2022, Frascati (RM), Italy
barbara.barricelli@unibs.it (B. R. Barricelli); daniela.fogli@unibs.it (D. Fogli)
https://barbara-barricelli.unibs.it/ (B. R. Barricelli); https://daniela-fogli.unibs.it/ (D. Fogli)
0000-0001-9575-5542 (B. R. Barricelli); 0000-0003-1479-2240 (D. Fogli)

In line with this technological evolution also the definition of EUD has to change; Barricelli et al. [15] proposed a revised definition that conceives EUD as "the set of methods, techniques, tools, and socio-technical environments that allow end users to act as professionals in those ICT-related domains in which they are not professionals, by creating, modifying, extending and testing digital artifacts without requiring knowledge in traditional software engineering techniques".

Recently, AI-infused systems are being introduced in users' everyday life, such as virtual assistants, collaborative robots, healthcare devices, autonomous trading systems, and drones. These systems may offer natural language interfaces, provide automatic reasoning, and show learning capabilities for adaptation to users' habits and preferences. However, users might need to keep control of systems and intervene to modify their behavior [16]. New methods and techniques for EUD can thus become more and more important in the AI age, to allow users not only to create and modify automations for their smart devices [17], but also trust in them [18].

In this position paper, we use two cases where AI is employed – virtual assistants and collaborative robots –, to explore how EUD could be of help, and how, vice versa, AI-infused systems may provide hints for an enhanced EUD paradigm. The final aim is to underline relevant opportunities for the future of EUD and the impact it might have on the quality of life of everybody.

2. Case 1: Virtual Assistants

Virtual Assistants (VAs) are AI-infused systems and their use for smart home control is becoming more and more popular. They have become well known thanks to the diffusion of several commercial physical devices that provide their own companion VA – e.g., Google Nest with Google Assistant, Amazon Echo with Amazon Alexa, and Apple HomePod with Siri.

A VA allows the user to interact with an IoT ecosystem [19] using a voice-based interaction to ask for information, to activate smart devices, and to communicate with other users who share the same environment.

Knowledge representation and machine learning techniques play an important role in VA use and operation, by providing users with contents and device behaviors tailored to their preferences and habits. At the same time, users can play an active role when interacting with an IoT ecosystem through VAs; in fact, they can shape and manage the ecosystem and its behaviour. This feature can be characterized as a form of EUD and is commonly enabled by applications that allow to define sequences of actions to be activated when specific events occur, i.e. *routines*.

In commercial VAs, a predefined set of routines is made available to the users; some examples are *Good Morning* or *I'm home*. Besides these, the users can create personalized routines through the VA mobile apps. A two-step generic procedure for routine creation, regardless the brand of the VA, can be defined as follows:

1. *Routine trigger definition*: the user defines how the routine will be started (with a direct voice command; at a specific time and date; at sunrise/sunset; when a sensor, connected to the VA, detects a specific event).
2. *Actions definition*: the user selects the actions to be executed when the routine is triggered. Some examples are: information gathered from specific providers (e.g., weather forecast,

traffic information), reminders (e.g., calendar events, shopping lists), announces (e.g., send messages, read incoming messages), commands to connected devices (e.g., light bulbs, electric plugs, thermostats), control of media sources (e.g., news, music player, radio stations).

In this specific case, AI and EUD could influence each other in different ways by opening up interesting research directions.

In current commercial VAs, routines can be created and modified only with the dedicated GUI-based companion app. However, being VAs able to perform Natural Language Processing (NLP), it would be interesting to design EUD techniques allowing users to manage IoT ecosystems and create routines just through voice commands and conversations with the VA [20]. In this way, NLP would be exploited to empower users in taking advantages of their assistants by improving usability and user experience. In addition, with the spread of VAs with embedded screens (e.g., Google Nest Hub or Amazon Echo Show), a multi-modal conversation-based paradigm could be conceived, which could better cope with the complexity and flexibility that some EUD tasks might require [21].

Automatic reasoning, planning algorithms, and machine learning methods could be adopted to provide the user with suggestions during routine creation; such suggestions can be derived from previous use of the virtual assistant and connected devices in the environment by the same user or by 'similar' users. Some initial attempts in this direction exploit the capabilities of recommeder systems [22].

EUD techniques have been designed and experimented till now by considering only single-user interaction with the EUD tool. But in the case of VAs, the scenario might change. VAs are usually installed in home environments, and therefore they are often available for a group of users (family members or roommates). Different types of EUD techniques might be offered to the users, in order to accommodate their different knowledge, skills and preferences. In addition, the creation of routines by one of the home inhabitants will be affecting the others and might be conflicting with routines created by others. This aspect has reflections on the complexity that EUD activities might have in multi-user and shared environments. In HCI, there exist some proposals that address this problem by sustaining collaboration among home inhabitants through gamification mechanisms [23]. Embracing AI in this context could instead yield more advanced behaviors: techniques based on uncertain reasoning and argumentation theory could be exploited by tailoring the interaction modality to the user's profile, by detecting possible conflicts and suggesting ways for their solutions [24, 25], and by recognizing in advance possible side-effects of a routine, thus proposing also in this case how to avoid them.

Viceversa, the EUD activities could inform AI-infused systems like VAs about the reasons of some user's choices, specifically the reasons of performing some actions in a particular order, linking them to a specific time of the day or to the occurrence of particular events. This in turn could lead to obtain proactive behaviors of VAs, such as suggestions about the routines that could be created or the order in which actions might be executed in a routine. In addition, if more than one user perform EUD activities for the same IoT ecosystem, it might happen that the VA could learn different users' preferences and adapt its suggestions to each of them.

3. Case 2: Collaborative Robots

Collaborative robots represent a promising technology for small-medium enterprises, where production is highly variable and characterized by small batches. Human workers and robotic devices may thus share the same space and collaborate in the accomplishment of tasks that require different types of competencies and abilities [26, 27]. Collaborative robots are endowed with safety mechanisms to prevent harming humans and should be easily programmed by human workers to cope with rapidly changing production needs. Furthermore, whenever robots show all these characteristics, they may work alongside people also in other spaces, such as houses, hospitals, shops or museums [28].

Several approaches to robot programming by end users have been proposed over the years: programming by demonstration [29], visual programming environments [30, 31], natural language interfaces [32], situated tangible programming [33], and hybrid programming [34]. These approaches exploit AI techniques for natural language understanding, image and gesture recognition, and arm movement reproduction. Therefore, collaborative robots and environments for robot programming can be regarded as AI-infused systems, which must be used by non-expert programmers. The latter are called on to define tasks to be performed by or with the robot, which usually refer to operations the robot must perform (e.g., pick-and-place, assemble, screw), the objects to be manipulated, and the locations where the objects can be found or put.

In this case, AI may inform EUD through the recognition of objects in the environment, by suggesting their use in the new task that the user is defining, or it may find patterns in sequences of operations the user required the robot to perform, by suggesting possible plan optimizations. As a consequence, the EUD environment designed to support operators in robot task programming could be progressively enriched, thanks to AI features, such as pattern recognition, knowledge representation and planning algorithms. AI may also support the operator in defining correct and safe programs: non-expert programmers could pay more attention to some aspects of the tasks, while neglecting important actions that are needed to make a task fully compliant with safety regulations; the lack of these actions could be recognized by AI algorithms and suggestions for program modification could be automatically provided.

Different types of operators at the shop floor might need to collaborate with a robot according to their physical characteristics, skills and aims; they can also have a different expertise in robot programming and thus require different styles of interaction to perform EUD activities (natural language, visual programming, rule-based interaction, or others). AI may help coping with this issue by recognizing the operators and their characteristics and proposing them the preferred EUD modality.

On the other hand, EUD may enable AI technologies in a sort of mixed-initiative approach: recognizing natural language commands or human gestures can be facilitated by the interaction with the operator, who might play the role of robot trainer during their EUD activity. In general, the EUD environment could provide functionalities for enriching the vocabulary of vocal commands or gestures on which the AI technologies are based. Through such an environment, each operator may also share programs with other users, and AI can sustain the re-use of existing solutions by proactively proposing them to the users when a known situation or problem is recognized.

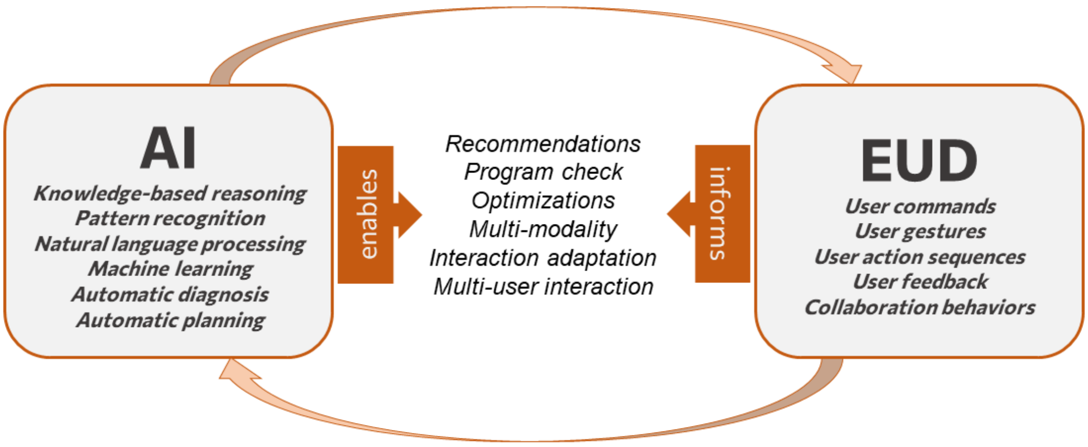

Figure 1: The reciprocal influence of AI and EUD

4. Discussion

As shown in the previous sections, novel application domains such as virtual assistants for IoT and collaborative robotics provide several opportunities for EUD and AI cross-fertilization. On the one hand, conceiving AI as a "tool" for EUD may help developing advanced EUD methods and techniques that could facilitate user's work during digital artifact adaptation and creation. When non-trivial EUD activities must be performed to obtain some desired behavior by a system, AI features could provide personalized suggestions, intuitive interaction modalities and active control on correctness and safety of user-defined artifacts. On the other hand, users should be able to understand and keep control on the behavior of AI-infused systems, in order to accept and appropriate it; EUD features can provide users with the means to manage such systems in a more personal and democratic way, by informing at the same time the AI algorithms about users' habits and preferences.

The reciprocal influence of AI and EUD is depicted in Figure 1: AI influences EUD by means of different AI algorithms (those listed in the figure are only some examples) while EUD influences AI by exploiting the activity history of the users (related to individual and collaborative interaction). This cross-feeding process allows AI to enable, and EUD to inform, fundamental features of modern AI-based systems: recommendations, program check, optimization, multi-modality, interaction adaptation, and multi-user interaction.

The above considerations yield some open challenges that would require further research to be properly investigated.

1. *How to support the user in becoming an end-user developer by means of AI algorithms?* E.g.: How can the routine creation be informed directly by the VAs, suggesting for example a specific setting for a service that users might like because they used it frequently? How can robot task programming be informed by AI features able to check program correctness and safety, or suggest possible optimizations?

2. *How can the EUD activity be used to inform the AI-based system about user's preferences, habits, and needs?* E.g.: How can routine creation or robot task programming provide information to enrich the AI-based system and make it able to provide better suggestions and recommendations in the future?
3. *How can interaction with EUD environments be fostered by AI algorithms?* E.g.: How can NLP be used to make routine creation more intuitive and engaging? How can object, speech and gesture recognition be exploited to facilitate robot programming?
4. *How can AI help with customization of multi-user and shared environments?* E.g.: How can a negotiation process be put in place to achieve the best possible outcome for all the people involved with the same IoT ecosystem? How can the re-use of robot tasks be favored by context-awareness and situation understanding capabilities of the AI-based system?

5. Conclusion

Studying and realizing an effective interplay between EUD and AI could represent an interesting research area for the future. In particular, the design of EUD environments might require a paradigm change: EUD will not aim only to support users in customizing traditional interactive systems but also in shaping the collaboration between humans and AI-based systems. New EUD methods and techniques could be used to actively involve stakeholders in the design and development of AI-based systems. This may help end users better control and master these systems, thus increasing acceptance and trust. On the other hand, AI algorithms could be implemented in EUD environments to make EUD activities easier and engaging for end users. We claim that a good balance between AI and EUD features, beyond the traditional trade-off between adaptation and adaptability, could help designing systems with a positive impact on the users' quality of life.

References

[1] H. Lieberman, F. Paternò, V. Wulf, End User Development (Human-Computer Interaction Series), Springer-Verlag, Berlin, Heidelberg, 2006.

[2] M. F. Costabile, D. Fogli, P. Mussio, A. Piccinno, End-user development: The software shaping workshop approach, in: H. Lieberman, F. Paternò, V. Wulf (Eds.), End User Development, Springer Netherlands, Dordrecht, 2006, pp. 183–205. doi:10.1007/1-4020-5386-X_9.

[3] G. Stevens, G. Quaisser, M. Klann, Breaking it up: An industrial case study of component-based tailorable software design, in: H. Lieberman, F. Paternò, V. Wulf (Eds.), End User Development, Springer Netherlands, Dordrecht, 2006, pp. 269–294. doi:10.1007/1-4020-5386-X_13.

[4] M. Won, O. Stiemerling, V. Wulf, Component-based approaches to tailorable systems, in: H. Lieberman, F. Paternò, V. Wulf (Eds.), End User Development, Springer Netherlands, Dordrecht, 2006, pp. 115–141. doi:10.1007/1-4020-5386-X_6.

[5] M. Burnett, G. Rothermel, C. Cook, An integrated software engineering approach for end-user programmers, in: H. Lieberman, F. Paternò, V. Wulf (Eds.), End User Development, Springer Netherlands, Dordrecht, 2006, pp. 87–113. doi:10.1007/1-4020-5386-X_5.
[6] J. F. Pane, B. A. Myers, More natural programming languages and environments, in: H. Lieberman, F. Paternò, V. Wulf (Eds.), End User Development, Springer Netherlands, Dordrecht, 2006, pp. 31–50. doi:10.1007/1-4020-5386-X_3.
[7] G. Fischer, E. Giaccardi, Meta-design: A framework for the future of end-user development, in: H. Lieberman, F. Paternò, V. Wulf (Eds.), End User Development, Springer Netherlands, Dordrecht, 2006, pp. 427–457. doi:10.1007/1-4020-5386-X_19.
[8] C. S. De Souza, S. D. J. Barbosa, A semiotic framing for end-user development, in: H. Lieberman, F. Paternò, V. Wulf (Eds.), End User Development, Springer Netherlands, Dordrecht, 2006, pp. 401–426. doi:10.1007/1-4020-5386-X_18.
[9] H. Lieberman, F. Paternò, V. Wulf, End User Development (Human-Computer Interaction Series), Springer-Verlag, Berlin, Heidelberg, 2006.
[10] F. Paternò, V. Wulf (Eds.), New Perspectives in End-User Development, Springer, Cham, 2017. doi:10.1007/978-3-319-60291-2.
[11] C. Ardito, P. Buono, G. Desolda, M. Matera, From smart objects to smart experiences: An end-user development approach, International Journal of Human-Computer Studies 114 (2018) 51–68. doi:https://doi.org/10.1016/j.ijhcs.2017.12.002.
[12] D. Caivano, D. Fogli, R. Lanzilotti, A. Piccinno, F. Cassano, Supporting end users to control their smart home: design implications from a literature review and an empirical investigation, Journal of Systems and Software 144 (2018) 295–313. doi:https://doi.org/10.1016/j.jss.2018.06.035.
[13] F. Corno, L. De Russis, A. M. Roffarello, Heytap: Bridging the gaps between users' needs and technology in if-then rules via conversation, in: Proceedings of the International Conference on Advanced Visual Interfaces, AVI '20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 1–9. doi:10.1145/3399715.3399905.
[14] F. Paternò, C. Santoro, End-user development for personalizing applications, things, and robots, International Journal of Human-Computer Studies 131 (2019) 120–130. doi:https://doi.org/10.1016/j.ijhcs.2019.06.002.
[15] B. R. Barricelli, F. Cassano, D. Fogli, A. Piccinno, End-user development, end-user programming and end-user software engineering: A systematic mapping study, Journal of Systems and Software 149 (2019) 101–137. doi:https://doi.org/10.1016/j.jss.2018.11.041.
[16] A. Schmidt, T. Herrmann, Intervention user interfaces: A new interaction paradigm for automated systems, Interactions 24 (2017) 40–45. doi:10.1145/3121357.
[17] F. Paternò, When artificial intelligence alone is not enough: End-user creation and control of daily automations, in: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, New York, NY, USA, 2021, pp. 1–3. URL: https://doi.org/10.1145/3411763.3445005.
[18] B. Shneiderman, Bridging the gap between ethics and practice: Guidelines for reliable, safe, and trustworthy human-centered ai systems, ACM Trans. Interact. Intell. Syst. 10 (2020). doi:10.1145/3419764.
[19] B. R. Barricelli, E. Casiraghi, S. Valtolina, Virtual assistants for end-user development in the internet of things, in: A. Malizia, S. Valtolina, A. Morch, A. Serrano, A. Stratton (Eds.),

End-User Development, Springer International Publishing, Cham, 2019, pp. 209–216.
[20] B. R. Barricelli, D. Fogli, Virtual assistants for personalizing iot ecosystems: Challenges and opportunities, in: CHItaly 2021: 14th Biannual Conference of the Italian SIGCHI Chapter, CHItaly '21, Association for Computing Machinery, New York, NY, USA, 2021, pp. 1–5. doi:10.1145/3464385.3464699.
[21] B. R. Barricelli, D. Fogli, L. Iemmolo, A. Locoro, A multi-modal approach to creating routines for smart speakers, in: Proceedings of the 2022 International Conference on Advanced Visual Interfaces (AVI 2022), Association for Computing Machinery, New York, NY, USA, 2022, pp. 1–5. doi:10.1145/3531073.3531168.
[22] F. Corno, L. De Russis, A. M. Roffarello, From users' intentions to if-then rules in the internet of things, ACM Trans. Inf. Syst. 39 (2021). doi:10.1145/3447264.
[23] F. Benzi, F. Cabitza, D. Fogli, R. Lanzilotti, A. Piccinno, Gamification techniques for rule management in ambient intelligence, in: B. E. R. de Ruyter, A. Kameas, P. Chatzimisios, I. Mavrommati (Eds.), Ambient Intelligence - 12th European Conference, AmI 2015, Athens, Greece, November 11-13, 2015, Proceedings, volume 9425 of *Lecture Notes in Computer Science*, Springer, 2015, pp. 353–356. doi:10.1007/978-3-319-26005-1_25.
[24] K. Nurgaliyev, D. D. Mauro, N. Khan, J. C. Augusto, Improved multi-user interaction in a smart environment through a preference-based conflict resolution virtual assistant, in: 2017 International Conference on Intelligent Environments (IE), 2017, pp. 100–107. doi:10.1109/IE.2017.21.
[25] B. Ospan, N. Khan, J. Augusto, M. Quinde, K. Nurgaliyev, Context aware virtual assistant with case-based conflict resolution in multi-user smart home environment, in: 2018 International Conference on Computing and Network Communications (CoCoNet), 2018, pp. 36–44. doi:10.1109/CoCoNet.2018.8476898.
[26] J. Colgate, W. Wannasuphoprasit, M. Peshkin, Cobots: robots for collaboration with human operators, in: Proceedings of the 1996 ASME International Mechanical Engineering Congress and Exposition, 1996, pp. 433–439.
[27] V. Villani, F. Pini, F. Leali, C. Secchi, Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications, Mechatronics 55 (2018) 248–266. doi:https://doi.org/10.1016/j.mechatronics.2018.02.009.
[28] L. Rozo, S. Calinon, D. G. Caldwell, P. Jiménez, C. Torras, Learning physical collaborative robot behaviors from human demonstrations, IEEE Transactions on Robotics 32 (2016) 513–527. doi:10.1109/TRO.2016.2540623.
[29] S. Alexandrova, M. Cakmak, K. Hsiao, L. Takayama, Robot programming by demonstration with interactive action visualizations, in: Robotics: Science and Systems, 2014, pp. 1–9.
[30] J. Huang, M. Cakmak, Code3: A system for end-to-end programming of mobile manipulator robots for novices and experts, in: 2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI), 2017, pp. 453–462.
[31] D. Weintrop, A. Afzal, J. Salac, P. Francis, B. Li, D. C. Shepherd, D. Franklin, Evaluating CoBlox: A Comparative Study of Robotics Programming Environments for Adult Novices, Association for Computing Machinery, New York, NY, USA, 2018, p. 1–12.
[32] D. K. Misra, J. Sung, K. Lee, A. Saxena, Tell me dave: Context-sensitive grounding of natural language to manipulation instructions, The International Journal of Robotics Research 35 (2016) 281–300. doi:10.1177/0278364915602060.

[33] Y. S. Sefidgar, P. Agarwal, M. Cakmak, Situated tangible robot programming, in: Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction, HRI '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 473–482. doi:10.1145/2909824.3020240.

[34] D. Fogli, L. Gargioni, G. Guida, F. Tampalini, A hybrid approach to user-oriented programming of collaborative robots, Robotics and Computer-Integrated Manufacturing 73 (2022) 102234. doi:https://doi.org/10.1016/j.rcim.2021.102234.