

From models to impact: a human-AI approach for effective support of collaborative teams in agile product development

Roman Ognevoj^{1,✉}, Anusch Musawi², Milenko Bugueno¹, Marcel Busch², Mona Batora², Katharina Ritzer², Nikola Bursac² and Kristin Paetzold-Byhain¹

¹ Chair of Virtual Product Development, Dresden University of Technology, Germany,

² ISEM - Institute for Smart Engineering and Machine Elements, Hamburg University of Technology, Germany

✉ roman.ognevoj@tu-dresden.de

ABSTRACT: Agile teams often encounter obstacles in systematically identifying the underlying root causes of collaboration challenges and deriving effective countermeasures. Grounded in the Design Research Methodology, this study investigates a hybrid AI-human approach for targeted generation of problem-specific reference and impact models to enhance systematic improvement in agile product development. A structured workflow integrates AI capabilities (e.g. scaffolding, consistency) and expert knowledge (causality, context), while a three-stage review ensures methodological rigor and result reliability.

KEYWORDS: agile development, artificial intelligence (AI), network analysis, reference model, impact model

1. Introduction and motivation

In dynamic and uncertain development environments, it is crucial for organizations to maintain responsiveness (Sutherland & Sutherland, 2014). In the context of prevailing VUCA conditions, development teams are required to respond quickly and flexibly to change (Pendzik et al., 2023). This necessity underlies the frequent integration of agile frameworks into development processes (Scharold & Paetzold-Byhain, 2024). However, practice shows that implementation in isolation is insufficient to overcome challenges when the underlying mechanisms and intentions of agile practices remain poorly understood (Suryaatmaja et al., 2020). Effectiveness emerges only when goals, purposes, and mechanisms of action are known and aligned with the specific challenge at hand (Suryaatmaja et al., 2020; Sutherland & Sutherland, 2014).

In order to implement agile frameworks in a targeted and supportive manner within the context of team collaboration, it is essential to identify and address the root cause of a given challenge. In practice, however, it can be observed that agile teams often encounter difficulties in identifying the specific root causes of challenges (Ciancarini et al., 2021). Instead, they tend to describe only the absence of a desired target state. Consequently, the derivation of effective support measures is complicated by two factors. First, the poor quality of the problem definition, and second, the lack of knowledge about relevant influencing factors and underlying mechanisms (Ciancarini et al., 2021; Ferreira et al., 2021). In order to develop targeted interventions and address issues at the appropriate level rather than symptoms, it is therefore necessary to have a precise depiction of the current state (Blessing & Chakrabarti, 2009).

To address this problem, this paper proposes a Human-AI approach intended to assist development teams in dealing with such challenges. This approach is grounded in the work of Batora et al. (2025), which outlines a model that captures relevant influencing factors, relationships, and mechanisms of agile team collaboration. The rationale for this Human-AI approach lies in the complementarity of capabilities: while the approach leverages LLMs for the scalability required for structural analysis and model

scaffolding across complex datasets, human expertise remains indispensable for causal and contextual interpretation, evaluation, and validation of these structures. Based on this foundation, teams are supported in representing both the current state (Reference Model) and associated influencing factors, as well as the corresponding desired state (Impact Model) in a problem-specific manner. By coupling the LLM's efficiency with the contextual validity of human expert judgment, the approach ensures that derived models are both structurally sound and contextually valid. The long-term objective is to enable needs-based, problem-specific support in order to improve team collaboration within the context of agile product development.

2. State of research

2.1. Problems & challenges in agile collaboration

Agility is generally defined as an organization's ability to initiate and respond to change quickly, while learning in the process and generating customer value at the same time (Conboy, 2009). It provides the cultural and structural foundation for managing uncertainty and complexity in dynamic development environments (Pendzik et al., 2023). When applied correctly, agile frameworks such as Scrum encourage iterative learning, goal adaptability, responsiveness to environmental changes, and transparent collaboration within and between teams (Baham & Hirschheim, 2022). In practice, however, organizations often fail to realize these and other intended benefits. When applying agile frameworks, development teams frequently encounter numerous challenges, as agile frameworks such as Scrum or Kanban are often adopted more in form than in spirit and therefore not in a way that aligns with their underlying purposes. For example, when communication problems arise, teams often introduce daily scrum meetings to improve information flow (Pikkarainen et al., 2008). However, if these meetings are not used for their intended purpose of providing a short, structured, and focused exchange of information, or if the measure does not address the actual root cause (e.g. a lack of trust within the team), the situation may deteriorate. This can create additional issues, such as time loss due to non-value-adding activities (Stray et al., 2020). Unnecessary meetings can also generate a high volume of unfiltered information, contributing to information overload, fragmented situational awareness, and reduced decision quality (Pikkarainen et al., 2008). Another critical challenge is, for example, the high prevalence of task switching and frequent interruptions (Abad et al., 2018). These frequent disturbances, whether internal or external, disrupt cognitive focus, increase switching costs, and significantly impair task performance (Abad et al., 2018). In agile environments characterized by parallel work streams and continuous coordination demands, a high frequency of interruptions can lead to sustained cognitive fragmentation and diminish overall team effectiveness. In order to provide targeted and effective support to teams through agile practices, it is therefore necessary to identify the root cause of a given challenge. However, without a clear identification of the underlying causes and a systematic understanding of the mechanisms that shape agile team collaboration, improvement measures often remain generic and short-lived (Baham & Hirschheim, 2022; Stray et al., 2020).

2.2. Large language models

To address this need for systematic understanding through analyzing mechanisms of agile team collaboration, Large Language Models (LLMs) represent a powerful support tool for data-driven analysis and modeling in product development (Chiarello et al., 2024). Based on transformer architectures, they capture complex textual and semantic relationships and derive structured knowledge representations (Bommasani et al., 2022; Chiarello et al., 2024). This makes them particularly suitable for tasks where unstructured descriptions of collaboration problems, mechanisms, and contextual factors must be translated into relational models. Classical data-driven relation extraction methods - such as distant supervision or convolutional neural networks - provide robust and reproducible baselines (Zeng et al., 2014), but often struggle with the unambiguous mapping of extracted relations to a predefined knowledge base. LLMs complement these approaches by generalizing from few examples, generating context-rich hypotheses about causal paths or role functions in teams, and flexibly adapting to domain-specific terminologies (Bommasani et al., 2022; Di Rocco et al., 2025). At the same time, models with high generative capacity are more susceptible to hallucinations and inconsistencies (Maynez et al., 2020), whereas more constrained models tend to prioritize controllability and traceability. In both cases, safeguards such as retrieval-augmented generation and systematic human review remain essential to

ensure factuality and methodological coherence (Lewis et al., 2020), which also supports the necessity of the two-part nature of the Human-AI approach presented in this paper. In the context of network analysis, LLMs do not replace established structural metrics such as degree, betweenness, or modularity (Newman, 2018), but add a semantic layer to them. They can assign meaningful labels to nodes and edges, extract topics and intentions from textual data, and suggest plausible influence paths at the level of mechanisms (S. Wang et al., 2025). This linkage of qualitative insights with quantitative network signals substantially increases the action relevance of the resulting models.

From a machine learning perspective, classical statistical and supervised learning methods remain well suited for identifying numerical correlations and predicting scalar outcomes from structured data (Bishop, 2006; Murphy, 2012). However, the derivation of domain-specific causal models from textual and semi-structured sources requires high-capacity semantic reasoning across heterogeneous inputs (Bommasani et al., 2022). LLMs address this gap by functioning as cognitive scaffolds that transform unstructured or semi-structured knowledge into coherent relational representations (Chiarello et al., 2024; Zhou & Chen, 2025). Within this role, they provide the semantic and structural scaffolding, while human experts retain authority over the final causal interpretation and integration into the overall design research methodology.

3. Research gap & target

As discussed in Section 1, the effective implementation and application of agile frameworks can prove to be problematic (Suryaatmaja et al., 2020; Sutherland & Sutherland, 2014). In practice, the identification and precise definition of the origin of an observed problem often proves difficult, as root causes are not immediately evident and additional contributing factors and their underlying mechanisms are likewise not readily identifiable (Ciancarini et al., 2021; Ferreira et al., 2021). An imprecise problem definition can significantly hinder the identification and implementation of appropriate countermeasures (Blessing & Chakrabarti, 2009; Suryaamaja et al., 2020). To address these challenges in a systematic and efficient manner and to support teams in agile product development, a Human-AI approach is introduced. This approach builds on the research of Batora et al. (2025), which offers an integrative and validated Main Model (MM) capturing different influencing factors, causal mechanisms of action, and interdependencies affecting team collaboration in agile product development. The Human-AI approach leverages the MM as an initial knowledge base to support teams in accurately mapping their problem-specific current state (Reference Model), identifying key influencing factors as root causes for observed problems, and deriving a corresponding desired state (Impact Model) according to the concept of Blessing and Chakrabarti (2009). LLMs within this approach are not proposed as a standalone modeling methodology. Instead, they function as an advanced support tool to operationalize the initial scaffolding of Reference and Impact Models. While the LLM is capable of providing the analytical capacity to navigate and synthesize the extensive knowledge base of the MM, the efficacy of the procedure is complemented and validated by human contextualization, causal assessment, and refinement.

In doing this the Human-AI approach is intended to contribute to achieving the overarching objective of enhancing team collaboration and enabling a more effective and goal-oriented utilization of agile frameworks in dynamic and uncertain development environments.

From these considerations, the following research questions are proposed:

1. How can an approach that combines AI and human expertise efficiently support the development of problem-specific Reference and Impact Models in agile product development?
2. How can potentially effective support measures be derived from the hybrid approach and successfully implemented in practice?

3.1 Research approach

To systematically address the research questions, the methodological approach follows the Design Research Methodology (DRM) (Blessing & Chakrabarti, 2009), enabling the structured development of prescriptive support based on an identified research gap and an existing descriptive knowledge base. As the Reference Model and the Impact Model play a central role in this contribution, the DRM provides the necessary framework for analyzing design processes through networks of influencing factors and interdependencies. Building on this framework, the objective of this work is to utilize the current MM - which includes 148 influencing factors and 333 causal connections, synthesize insights from academic

literature and expert input, and provides a foundational framework for addressing challenges in specific development contexts - as a structured knowledge base for developing targeted, problem-specific Reference and Impact Models.

In this context, the 'Reference Model' (RM) captures the current state of a development situation through relevant influencing factors and their causal relationships. The 'Impact Model' (IM) defines the desired state and anticipated improvements (Success Factors) through targeted support. Within both models, Key Factors and Success Factors are defined: 'Key Factors' represent critical influences that act as root causes for support interventions, while a 'Success Factor' is located at the end of the cause-effect chain and serves as the target factor, which is indirectly addressed through the causal interdependencies influenced by support measures linked to a chosen Key Factor (Blessing & Chakrabarti, 2009). To operationalize this procedure, the MM was initially visualized and extended using the graphical user interface of Kumu (kumu.io). However, as the number of factors and connections grew, maintaining and navigating the model became increasingly complex. This complexity provided a primary rationale for designing the Human-AI approach, which leverages Coding Agents (e.g. OpenAI Codex, Claude Code, GitHub Copilot) and LLMs (e.g. GPT 5, Claude 4, Perplexity, Grok 4) to support the model development, manage growing complexity, synthesize extensive knowledge structures efficiently and facilitate scaffolding processes. To operationalize this approach, the MM was exported as a JSON file - including all influencing factors, along with their connections and attributes - and version-controlled in an open-source GitHub repository. To effectively leverage the potential of AI, three main use cases were defined with specific validation criteria: 1) adapting and evolving the MM by using a Coding Agent; 2) scaffolding a problem-specific RM using an LLM; 3) scaffolding an IM using an LLM and deriving support measures for practical implementation.



Figure 1. Main workflow for model development

A comprehensive review and validation process was established for the three main use cases. The underlying causal relationships and the workflow are illustrated in Figure 1. The Coding Agent initially performs a task, such as evaluating whether a new influencing factor meets Gatekeeping Criteria for inclusion in the MM. The expert then assesses adherence to criteria and analyzes causality and context, followed by a peer review evaluating the collaborative outcome.

Accordingly, the objective of this research is to examine how a Human-AI approach can be structured to leverage the complementary strengths of both, in order to systematically support team collaboration in agile product development by enabling teams to derive problem-specific support measures and successfully implement them in practice.

4. Results

4.1. Human-AI workflow for model development

To address the first research question, the primary tasks required for the three main use cases of this hybrid approach, as described in Section "Research approach", were decomposed. This included searching for existing influencing factors, adding new influencing factors and connections, analyzing the adherence of factor formulations to the DRM, creating scripts for automatic schema validation, and generating Pull Requests to facilitate peer reviews. This structured decomposition enabled a systematic evaluation of how AI could complement human expertise in these processes.

Consequently, the designed workflow focuses on: 1) AI: Execute model development tasks end-to-end; 2) Human: Define goals, constraints, and review changes. The adopted interface for this interaction combines coding agents for model evolution and structured prompt interfaces for model scaffolding, primarily for design researchers as main users in the current research state, acting as model architects. Following a detailed analysis and review process, the Human-AI approach workflow depicted in Figure 2

was established. In order to enhance and more systematically regulate extensions to the MM, Gatekeeping Criteria were developed incrementally, guiding both AI and human efforts in task execution and validation, safeguarding model integrity, ensuring methodological consistency, and maintain compatibility with Kumu visualization and DRM principles.

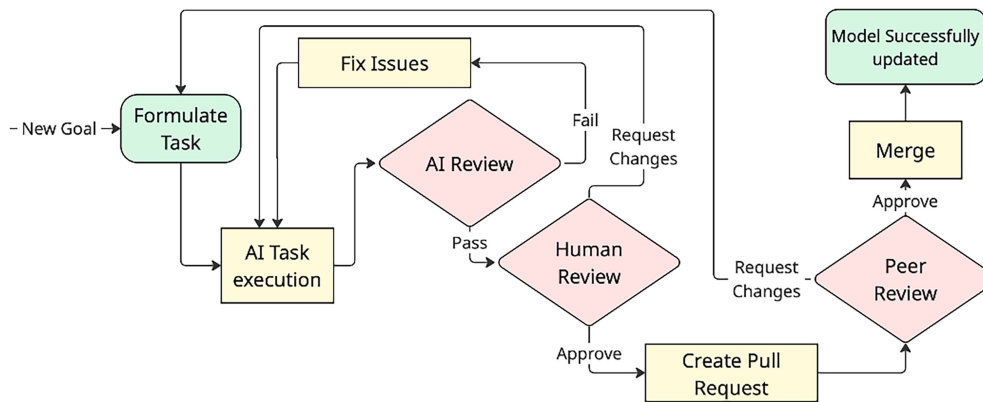


Figure 2. Human-AI workflow for model development

For instance, any proposed new influencing factor must meet the predefined criteria, according to [Blessing and Chakrabarti \(2009\)](#), including the right formulation (an attribute of an element), relevance, observability, measurability, and domain alignment. Proposed factors not meeting these standards are therefore systematically rejected. For model development tasks like introducing new influencing factors to the MM, Coding Agents have demonstrated impressive efficiency and effectiveness, particularly in breaking tasks into manageable sub-tasks. [Figure 3](#) illustrates the outcome for the task of inserting and evaluating a new influencing factor, “Quality of the air” (“Qualität der Luft”) by the Coding Agent. If both the Coding Agent and human reviewers approve the proposed factor, the Coding Agent integrates the changes into the MM and generates a Pull Request for final peer review and verification.

Criteria	Assessment	Status
DRM Methodology Compliance - Factor Formulation	Passed	✅ Correct "attribute-of-element" format
Scope & Domain Relevance	Rejected	❌ Outside design research methodology domain
Theoretical Foundation	Rejected	❌ Not aligned with design research theory
Causal Network Integration	Rejected	❌ Cannot meaningfully integrate into design process chains
Scientific Rigor Requirements	Rejected	❌ Lacks grounding in DRM/design research literature
Overall Verdict: REJECT - "Qualität der Luft" should not be included in the Main Model.		

Figure 3. Coding Agent verdict for a proposed influencing factor

Similar to the Gatekeeping Criteria, RM-Criteria and IM-Criteria were designed to systematically guide a coherent development of RM and IM, respectively. A critical requirement for RM-Criteria is the identification of a problem-specific Key Factor and a clear and well-defined cause-effect chain that links directly to the problem-specific Success Factor. The IM-Criteria specify the development of an IM, which is based on the validated RM, and an inclusion of an additional element, referred to as the support, which directly addresses the identified problem-specific Key Factor and contributes to achieving the desired Success Factor. This ensures a logical and coherent connection between the existing situation and the desired improved state. These Criteria are maintained in the repository to provide easy access, facilitate updates, and guarantee consistency throughout the process. For tasks such as scaffolding a problem-specific RM or scaffolding an IM to derive support measures for practical implementation, a Coding Agent has proven to be effective as well. However, for core Human-AI approach tasks, an isolated LLM was preferred due to intuitive and accessible interaction from a user perspective, as it avoids the technical complexity and orchestration overhead of the Coding Agent’s environment, and allows faster iterative prompt refinement with structured JSON inputs.

Targeted tests evaluated LLM effectiveness for scaffolding, focusing on content validity rather than quantitative benchmarks. Each model tested (ChatGPT 5, Claude Sonnet 4.5, Perplexity Pro, and Grok 4)

was evaluated based on its ability to comprehend the JSON MM structure, the plausibility of causal relationships, adherence to predefined RM-Criteria and IM-Criteria, and the frequency of content hallucinations. Perplexity Pro and Claude Sonnet 4.5 provided more conservative and source-oriented contributions, and were therefore considered complementary inputs, which aligns with existing findings on generative capability, hallucination risks, and the importance of source verification and relation extraction in unstructured data sources. In the qualitative assessment, ChatGPT 5 and Grok 4 interpreted the MM in a differentiated manner and established consistent contextual links. Ultimately, ChatGPT 5 was selected as the primary model for the approach, as it demonstrated the highest consistency in maintaining causal structures and adhering to the criteria, whereas Grok 4 exhibited a higher frequency of structural hallucinations.

4.2. Development of a reference and impact model

4.2.1. Evaluation of the required prompts

To develop the RM, a prompt was designed that derives a problem-specific model from the attached JSON MM. The prompt anchored the extraction in the relevant chapter on RMs from [Blessing and Chakrabarti \(2009\)](#) and required that mechanisms and model elements be taken directly from the JSON and aligned with that structure. As an initial test case, the problem of an “ineffective Daily Scrum” was selected. This scenario was chosen deliberately, as it represents a well-understood and densely connected knowledge cluster within the MM, ensuring traceability and comparability during evaluation.

The prompt was first executed using Grok 4. Although the model demonstrated adequate comprehension of the graph-like structure, the extracted RMs exhibited recurring issues, including hallucinated factors, inconsistent terminology, incorrect causal sequencing, and incorrect identification of a problem-specific Key Factor. Since one of the critical criteria for RM construction is the correct identification of a problem-specific Key Factor, these inconsistencies rendered Grok 4 unsuitable for further evaluation. The same prompt was subsequently tested with ChatGPT 5. In contrast to Grok 4, ChatGPT 5 produced substantially more coherent and consistent model structures. The identified mechanisms, their mapping to relevant nodes and edges, and the alignment with the established RM-Criteria demonstrated a satisfactory level of accuracy. Expert review confirmed that only minor adjustments were needed, likely due to the strong grounding of this problem domain within the MM. Notably, the LLM-generated RM was produced in under one minute, whereas manually deriving a comparable model from a network of more than 100 factors and over 300 connections would have required significantly more time and expert effort.

Building on the validated RM, the IM was generated using an analogous prompt structure that included the RM content, the corresponding chapter from Blessing and Chakrabarti, and the predefined IM-Criteria. The LLM produced a coherent IM within seconds, demonstrating consistent causal reasoning, clear justification of expected effects, and a structurally aligned derivation of intervention hypotheses. Expert evaluation again confirmed the substantive validity of the result. In summary, the “ineffective Daily” use case serves as an initial, controlled scenario to evaluate whether the LLM can correctly interpret the MM, derive a problem-specific causal chain, and produce a structurally sound RM and IM under well-understood conditions. This step establishes the basic functional validity of the prompt design and the Human-AI approach before applying it to team problems in agile product development identified in a survey ([Batora et al., 2025](#)).

4.2.2. Validation of the prompt with real team problem scenarios

The aim of the validation is to assess whether the prompt-supported method yields robust results in real application scenarios. Only if causal structure, measurability, and levers of intervention remain consistent in the usage context, the approach can be considered reliable and transferable. For the initial use case, problems identified in a survey previously discussed in the work of [Batora et al. \(2025\)](#) - such as “frequency of external interruptions during work” - were used for validation. The underlying RM provides the semantic frame by structuring the problem into start, intermediate, and end nodes with directed, annotated connections, including operationalizations and indications of influenceability. The RM generated by the LLM, together with the causal linkage extracted from the MM, is presented in [Figure 4](#) (left). In this representation, start nodes appear in yellow, causal paths between influencing factors in green and Success Factors in black. Factors that are measurable and influenceable are highlighted in magenta at both the node and edge level.

Building on this representation, the prompt consistently identifies the relevant mechanisms, the causal structure is substantively well-founded, and measurability as well as intervention levers are specified with sufficient precision. Furthermore, the resulting RM underwent expert review and subsequent peer review, which confirmed its suitability for the problem context. These conditions were fulfilled in the case of “Frequency of external interruptions during work.”

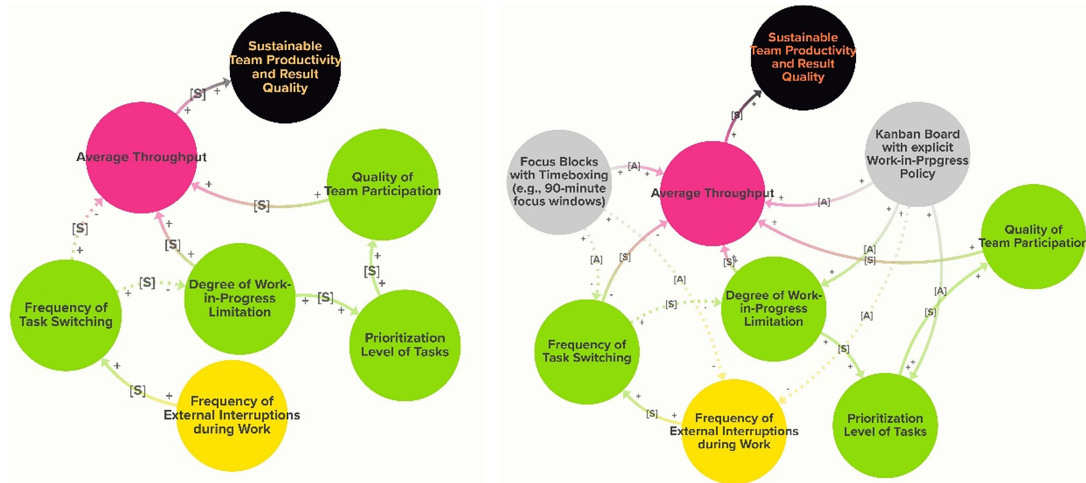


Figure 4. LM-based reference model (left) and Impact Model (right) addressing the specific issue of “frequency of external work interruptions”

Subsequently, the IM was generated. The IM prompt incorporates the JSON file of the RM, references the theoretical foundations for impact modeling as described by Blessing and Chakrabarti, and adheres to the previously defined IM-Criteria. The resulting IM structure displays the operational interventions (in gray) and their causal paths, such as focus blocks with time-boxing and Work-in-Progress policies, which act on the relevant start and intermediate nodes.

Execution with ChatGPT 5 produced consistent results in a very short time, which were confirmed in expert review and required only minor adjustments. In such cases, the necessary adjustments may include modifying the color representation of the factors that are both influenceable and measurable. For instance, it is possible both to influence and measure the frequency of task switching; therefore, the influencing factor should in fact be magenta instead of green. Although this is not reflected adequately in the color coding of the two models, the support measures in the IM are causally linked to it. This means that any change in this example would be limited to a visual adjustment. However, this was left out in order to provide a practical illustration of expert-driven adjustments.

This approach preserves the throughline: the RM serves as the current state of a development situation through relevant influencing factors and their causal relationships, and the IM translates this frame into concrete, measurable interventions for the survey-identified problem of “frequency of external interruptions during work”. In sum, the pipeline is well suited to practice settings where underlying causes are uncertain but deviations from the desired state are evident. From an observed problem, it reliably generates problem-specific RM and IM that (i) articulate a transparent causal narrative, (ii) translate this into measurable, intervention-ready levers, and (iii) remain auditable and refinable through expert review. Together, these qualities provide a defensible and efficient basis for selecting, testing, and iterating interventions to address practical challenges in team collaboration within the context of agile product development, leading toward durable improvement.

4.3. Support interventions and their practical implementation

In order to address the first part of the second research question, the previous sections have outlined how a problem-specific RM, and subsequently an IM incorporating the relevant support measures, can be generated from the MM by using LLM. The LLM generates an anticipated effective support measure grounded in the underlying knowledge base of the MM and the predefined constraints. The assumed effectiveness and the influenceability of relevant factors are justified through weighted relations within the MM. Figure 4 illustrates a generated IM with the aforementioned problem “frequency of external

interruptions during work” and the influence of the anticipated support interventions. For this problem, the LLM proposed two support measures that address the observed issue and the directly influenceable factors. Firstly, the implementation of focus blocks with timeboxing, i.e., team-wide, synchronized, time-limited rules for establishing undisturbed focus phases, has the potential to reduce external interruptions and also decrease task switching. This can increase the average throughput of completed work, both in a direct manner and additionally through the reduction of task switching. Consequently, this addresses the Success Factor, which may lead to an improvement in team productivity and outcome quality, as illustrated in [Figure 4](#). Secondly, the potential implementation of a Kanban board with an explicit WIP policy, which establishes a visible pull system with clear WIP limits and explicit blocker visualization, can directly address the problem as well as improve additional relevant factors. In this case as well, an increase in average throughput is also expected, which in turn is predicted to enhance the Success Factor, team productivity and outcome quality.

Subsequently, an expert review is conducted, in which the AI-generated proposals are evaluated conceptually, contextually, and causally and adjusted as needed. If adjustments are made, a formal peer review provides a collective assessment to conclude the process. The evaluation of anticipated effectiveness and potential suitability for practical implementation draws on individual and collective expert knowledge as well as theoretical foundations from the literature, including coordination theory ([Okhuysen & Bechky, 2009](#)), media richness theory ([Schmidt et al., 2017](#)), and the theory of complex adaptive systems ([X. Wang & Conboy, 2009](#)). These perspectives offer complementary explanatory approaches for mechanisms that can also be observed in agile team collaboration. The intended outcome of the second part of the second research question is an examination of how potential support measures can be effectively implemented in practice. Accordingly, if the LLM-generated proposals for support measures pass expert reviews, they can be implemented in practice and examined and evaluated in terms of applicability, effectiveness, and acceptance. For targeted implementation, nudge workshops ([Batora et al., 2025](#)) are one suitable option. The initial basis is formed by the LLM-generated support measures, which undergoes critical review and adaptation in collaboration with the designated industry partner. Consequently, this approach enables the execution of team-specific and effective modifications, with the potential to enhance acceptance, applicability, and implementability. This, in turn, can lead to an enhancement in the effectiveness of the support measures.

5. Summary and discussion

The present research demonstrates that a Human–AI approach can make a structured contribution to addressing a central challenge in agile product development. This challenge is centered on the difficulty of precisely identifying the underlying causes of collaborative problems and of deriving effective interventions from them. The findings demonstrate that the integration of AI, divided into Coding Agent and LLMs (e.g. scaffolding, structuring, consistency) and human expertise (e.g. contextualization, causal assessment, and validation) substantially facilitates the efficient generation of problem-specific models, the systematic exploration of relevant influencing factors, and the derivation of potentially effective support measures suitable for practical application as initial solutions. The two components prove to be complementary, and the multi-stage validation process is essential for correcting semantic inaccuracies and hallucinations and ensuring causal plausibility. However, the hybrid approach does not yield a generalizable set of interventions; rather, it provides a methodological process framework that allows for iterative adaptation, thereby enhancing flexibility and increasing the likelihood of successful implementation of the support measures in practice.

Different team cultures, organizational structures, and technical conditions may significantly affect the effectiveness of specific measures in practice. Accordingly, support measures must be adapted to the respective team context to enhance acceptance, applicability, and the ability to evaluate their actual effectiveness. The nudging workshops serve as a valuable co-design phase in which experts and practitioner teams jointly refine and customize the interventions.

It is also important to note that the quality of LLM-generated models is dependent on the maturity of the underlying MM. As this MM expands dynamically, there is a possibility that priorities, weightings, and resulting measures may shift until the MM reaches a sufficient level of maturity so that further changes no longer produce substantive fluctuations in the results. Importantly, as this model expands, it gains precision and quality, rather than losing them. The authors furthermore expect that a continually growing and maturing knowledge model will yield increasingly robust initial solutions and suggestions for

support measures. Future research should additionally investigate complementary methods - such as multi-domain matrices from graph theory - to strengthen the consistency of the MM and improve the derivation of potential support measures.

Despite these limitations, the approach exhibits substantial potential. It supports model development, problem definition, the identification of relevant Key Factors, the construction of an understanding of causes, causal relationships, and underlying mechanisms, as well as the transparent derivation of problem-specific support measures. The division of competencies between AI and humans increases the efficiency of model development, structures implicit expert knowledge and makes it explicit, facilitates the formulation of consistent hypotheses, and supports both data-driven and semantic explorations. Furthermore, the co-design phase for the implementation of the potential support measures in practice fosters a sense of ownership among practitioner teams, thereby enhancing their willingness to implement the proposed measures. Consequently, the approach can be understood as a methodological process framework for the systematic diagnosis and evidence-based advancement of team collaboration, rather than as an automated decision-making or optimization system.

6. Outlook

For future research, four points are of central importance: (1) the systematic investigation of the convergence and stability of the MM knowledge base, for example, through extended graph-theoretical methods; (2) large-scale case studies to assess external validity and scalability; (3) further development of model robustness through more reliable LLM mechanisms, the refinement of necessary boundary conditions and criteria, and the expansion of suitable validation metrics; and (4) the examination of the acceptance, applicability, and actual effectiveness of the proposed support measures in practice. In conclusion, research regarding the identified difficulties of collaborative teams in agile product development should advance the Human-AI approach so that its primary usability is not only limited to design researchers, but teams themselves can directly use this approach as a holistic tool to address the difficulties they encounter directly and quickly.

Acknowledgement

This research was funded by the German Research Foundation (DFG) within the project “Elaboration of the Causal Relationships of Agile Methods for Information Flow Design to Increase Flexibility and Efficiency in the Development of Mechatronic Systems” (Project Number: 548801836). The authors gratefully acknowledge the DFG for their financial support.

References

- Abad, Z. S. H., Noaen, M., Zowghi, D., Far, B. H., & Barker, K. (2018). Two Sides of the Same Coin: Software Developers' Perceptions of Task Switching and Task Interruption. *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018*, 175–180. <https://doi.org/10.1145/3210459.3214170>
- Baham, C., & Hirschheim, R. (2022). Issues, challenges, and a proposed theoretical core of agile software development research. *Information Systems Journal*, 32(1), 103–129. <https://doi.org/10.1111/isj.12336>
- Batora, M., Busch, M., Zimmermann, V., & Bursac, N. (2025). Effective nudging as a booster for agile collaboration in innovation. *International Journal of Innovation Management*, 29(09n10), 2540016. <https://doi.org/10.1142/S136391962540016X>
- Batora, M., Ognevoj, R., Musawi, A., Bugueno, M., Nikou, E. R. P., Busch, M., Ritzer, K., Paetzold-Byhain, K., & Bursac, N. (2025). Understanding the Influence Mechanisms in Agile Collaboration: A Network Analysis. [accepted].
- Bishop, C. M. (2006). Pattern recognition and machine learning. *Springer Google Scholar*, 2, 1122–1128.
- Blessing, L. T. M., & Chakrabarti, A. (2009). *DRM, a Design Research Methodology*. Springer. <https://doi.org/10.1007/978-1-84882-587-1>
- Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., Arx, S. von, Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., ... Liang, P. (2022). On the Opportunities and Risks of Foundation Models (arXiv:2108.07258). arXiv. <https://doi.org/10.48550/arXiv.2108.07258>
- Chiarello, F., Barandoni, S., Škec, M. M., & Fantoni, G. (2024). Generative large language models in engineering design: Opportunities and challenges. *Proceedings of the Design Society*, 4, 1959–1968.
- Ciancarini, P., Farina, M., Masyagin, S., Succi, G., Yermolaieva, S., & Zagvozkina, N. (2021). Root Causes of Interaction Issues in Agile Software Development Teams: Status and Perspectives. In K. Arai (Ed.), *Advances*

- in *Information and Communication* (Vol. 1364, pp. 1017–1036). Springer International Publishing. https://doi.org/10.1007/978-3-030-73103-8_74
- Conboy, K. (2009). Agility from first principles: Reconstructing the concept of agility in information systems development. *Information Systems Research*, 20(3), 329–354.
- Di Rocco, J., Di Ruscio, D., Di Sipio, C., Nguyen, P. T., & Rubei, R. (2025). On the use of large language models in model-driven engineering: J. Di Rocco et al. *Software and Systems Modeling*, 24(3), 923–948.
- Ferreira, B., Kalinowski, M., Gomes, M. V. D. C., Marques, M. C., Lopes, H., & Barbosa, S. D. J. (2021). Investigating Problem Definition and End-User Involvement in Agile Projects that Use Lean Inceptions. *XX Brazilian Symposium on Software Quality*, 1–10. <https://doi.org/10.1145/3493244.3493268>
- Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., & Rocktäschel, T. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474.
- Maynez, J., Narayan, S., Bohnet, B., & McDonald, R. (2020). On Faithfulness and Factuality in Abstractive Summarization. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 1906–1919). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2020.acl-main.173>
- Murphy, K. P. (2012). *Machine learning: A probabilistic perspective*. MIT press. [https://books.google.de/books?hl=de&lr=&id=RC43AgAAQBAJ&oi=fnd&pg=PR7&dq=Murphy,+K.+P.+\(2012\).+Machine+Learning:+A+Probabilistic+Perspective.+MIT+Press.&ots=unmv9FNo5c&sig=2PaFijjovomwGP9aFiATmLs3kk](https://books.google.de/books?hl=de&lr=&id=RC43AgAAQBAJ&oi=fnd&pg=PR7&dq=Murphy,+K.+P.+(2012).+Machine+Learning:+A+Probabilistic+Perspective.+MIT+Press.&ots=unmv9FNo5c&sig=2PaFijjovomwGP9aFiATmLs3kk)
- Newman, M. (2018). *Networks*. Oxford University Press.
- Okhuysen, G. A., & Bechky, B. A. (2009). 10 Coordination in Organizations: An Integrative Perspective. *Academy of Management Annals*, 3(1), 463–502. <https://doi.org/10.5465/19416520903047533>
- Pendzik, M., Sembdner, P., & Paetzold, K. (2023). Identification and classification of uncertainties as the foundation of agile methods. *Proceedings of the Design Society*, 3, 2165–2174. <https://doi.org/10.1017/pds.2023.217>
- Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., & Still, J. (2008). The impact of agile practices on communication in software development. *Empirical Software Engineering*, 13(3), 303–337. <https://doi.org/10.1007/s10664-008-9065-9>
- Scharold, F., & Paetzold-Byhain, K. (2024). Towards agile automotive development: Benefits, challenges and organizational changes. *Proceedings of the Design Society*, 4, 345–352. <https://doi.org/10.1017/pds.2024.37>
- Schmidt, T. S., Wallisch, A., Bohmer, A. I., Paetzold, K., & Lindemann, U. (2017). Media richness theory in agile development choosing appropriate kinds of prototypes to obtain reliable feedback. *2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, 521–530. <https://doi.org/10.1109/ICE.2017.8279930>
- Stray, V., Moe, N. B., & Sjöberg, D. I. K. (2020). Daily Stand-Up Meetings: Start Breaking the Rules. *IEEE Software*, 37(3), 70–77. <https://doi.org/10.1109/ms.2018.2875988>
- Suryaatmaja, K., Wibisono, D., Ghazali, A., & Fitriati, R. (2020). Uncovering the failure of Agile framework implementation using SSM-based action research. *Palgrave Communications*, 6(1), 8. <https://doi.org/10.1057/s41599-019-0384-9>
- Sutherland, J., & Sutherland, J. J. (2014). *Scrum: The art of doing twice the work in half the time* (First edition). Currency.
- Wang, S., Huang, J., Chen, Z., Song, Y., Tang, W., Mao, H., Fan, W., Liu, H., Liu, X., Yin, D., & Li, Q. (2025). Graph Machine Learning in the Era of Large Language Models (LLMs). *ACM Transactions on Intelligent Systems and Technology*, 16(5), 1–40. <https://doi.org/10.1145/3732786>
- Wang, X., & Conboy, K. (2009). Understanding agility in software development through a complex adaptive systems perspective.
- Zeng, D., Liu, K., Lai, S., Zhou, G., & Zhao, J. (2014). Relation classification via convolutional deep neural network. *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, 2335–2344. <https://aclanthology.org/C14-1220.pdf>
- Zhou, Y., & Chen, C.-H. (2025). Examining the Impact of Large Language Models on Design: Functions, Strengths, Limitations, and Roles. *Design and Artificial Intelligence*, 100017.