Evaluation of Failure Communication Strategies for Trust Repair in Human-AI Collaboration

Stina Klein, Alexandru Wurm, Elisabeth André, Matthias Kraus
Chair for Human-Centered Artificial Intelligence
Augsburg University
{stina.klein, alexandru.wurm, elisabeth.andre, matthias.kraus}@uni-a.de

Abstract

The increasing application of Large Language Models (LLMs) in everyday tasks and at work highlights the crucial importance of trust in human-AI collaboration, particularly when an AI system fails. This paper investigates the effectiveness of failure communication strategies for trust repair in collaborative physical tasks involving a a chat-based AI assistant. A controlled experiment in which participants built LEGO cars guided by an LLM-based AI Assistant was used to evaluate whether findings from trust repair in a virtual environment, such as chatbots, translate to an environment comprising tangible tasks, and whether the timing of trust repair influences the outcome. Results indicate that actively communicating mistakes significantly improves trust compared to a no repair strategy, and that early repair tends to be more effective, indicating that failure communication, independent of the timing, is important for an appropriate calibration of trust.

Keywords: Failure Communication, Trust Repair, Human-AI Collaboration

1. Introduction

Trust is a fundamental social construct that has been extensively studied for decades in the social sciences. It can be understood as the belief in another party's reliability, goodwill, and intentions, despite potential risks and unexpected outcomes (Rotter, 1980; Mayer et al., 1995). This confidence forms the cornerstone of interpersonal relationships. In the field of human-computer interaction (HCI), trust also plays a central role, as machines increasingly resemble social actors and elicit similar social responses from users (Nass et al., 1994). As automation and conversational AI advance, particularly in the form of LLM-based agents such as ChatGPT, machines are now capable of assisting with complex tasks, further elevating the importance of trust in HCI (Schaefer et al., 2016). In this context, understanding the facets of trust in automation becomes essential. Studies by Muir and Moray (1996) found that trust in automated systems is based on factors such as perceived reliability, competence, and integrity. Moreover, users may exhibit under-reliance (i.e., using a system less than they ideally should) or over-reliance (i.e., becoming overly dependent on the system and accepting its outputs without sufficient critical thinking) based on their perception of system trustworthiness (Parasuraman and Riley, 1997). It has been shown, that a system's performance-related aspects, such as failures and accuracy, have a strong impact on the human-AI trust relationship (De Visser et al., 2020; Hancock et al., 2011). In human-to-human interaction, how failures are communicated and how individuals recover from them play a significant role in repairing trust (e.g., (Shapiro et al., 1992; Goffman, 2017; Quinn et al., 2017)). These principles are now being explored in human–AI interaction, to determine how AI should communicate its failures and to evaluate which strategies are most effective in regaining user trust (Baker et al., 2018; De Visser et al., 2018). For example, Esterwood and Robert (2022) examined the use of apologies with remorse after speech recognition errors in interactions between humans and robots. Trust was first established through a trust game in which the robot answered questions correctly, followed by a misrecognition of the user's speech. Results showed a trend towards restoring user trust. Similarly, Lee et al. (2010) analyzed trust in human-robot interactions and found that apologizing and were most effective in increasing people's willingness to use the service again. They also suggested that tailoring recovery strategies to personal preferences could amplify effectiveness.

While much research focuses on regaining user trust after mistakes, another line of work seeks to minimize failures in the first place by improving model performance. For instance, Wang and Li (2023) introduced the SALAM framework, a cooperative study assistant that enables LLMs to learn from mistakes by providing feedback. Although effective, it still relies on ground-truth comparisons, which are insufficient for complex reasoning tasks involving multiple intermediate steps. Another approach by Zhang et al. (2024) explores in-context learning (ICL), in the form of few-shot prompting, and introduces the LEAP framework, which focuses on teaching generalizable principles to avoid mistakes. Despite these advances, failures cannot be completely eliminated. Therefore, methods for

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 2942–2951
11–16 May 2026.

repairing trust after the occurrence of a failure remain imperative. The central aim of this work is to evaluate whether findings from dialogue-based trust repair also apply in a *physical collaboration* setting. Specifically, this work investigates whether the presence or absence of failure communication influences trust in AI during collaborative manufacturing, and whether the timing of trust repair affects its success in a physical environment.

To address this, a controlled experimental setup is designed in which participants build LEGO cars guided by an AI Assistant implemented using a dual-LLM system. Participants' trust in the system is measured under different conditions of failure communication and timing. The results of this study aim to deepen understanding of trust dynamics in human–AI collaboration, particularly when interactions include a tangible, physical component. Our results show that Communicating and repairing failures reliably improved collaboration: both early and late repair increased trust and satisfaction compared to offering no repair. In our short-term task scenario, whether an failure is acknowledged mattered more than when. Participants reported more issues when no repair was offered, while early repair elicited the fewest noticed problems and some positive remarks. Based on this we provide design implications for failure communication and rectification in human-AI collaboration.

2. Related Work

2.1. Trust in Human-AI Collaboration

Trust between humans has been defined in many ways throughout time. For instance, Rotter defined interpersonal trust as "expectancy held by an individual that the word, promise, or written communication of another can be relied upon" (Rotter, 1967). Similarly, Mayer et al. described it as "the willingness of a party to be vulnerable to the outcomes of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to monitor or control that other party" (Mayer et al., 1995). Other works argue that trust is multidimensional, with trust and distrust forming separate constructs rather than lying on a single spectrum (Lewicki et al., 1998; Luhmann, 2018). By contrast, trust in technologies, robots, or AI appears to be more one-dimensional and typically concerns the human's expectations of the system and how that system will perform (Lee and See, 2004). Typically, it is measured using psychologically validated scales, such as the trust in automation scale by (Jian et al., 2000), which also has been applied often in previous works of human-AI collaboration (e.g. see Kraus et al. (2020)).

2.2. Failures in Human-AI Collaboration

Research on human–AI collaboration has increasingly moved beyond raw model accuracy to examine how failures shape team outcomes and trust. We adopt a pragmatic view in which a *failure* denotes a divergence between the team's output and a normative or task ground truth (irrespective of source), while a *mistake* denotes a human action or decision that is incorrect given available information (e.g., accepting a wrong AI suggestion or rejecting a correct one). This distinction aligns with work that separates *system errors* from *use errors* and shows that collaboration introduces additional failure modes beyond those of either party alone (Green and Chen, 2019).

A consistent finding is that people struggle to calibrate reliance on AI. Studies document over-reliance, e.g., errors of commission or accepting incorrect AI advice, and under-reliance, e.g., errors of omission or dismissing correct AI advice, as primary collaboration-specific mistakes that degrade team performance and trust (Bansal et al., 2021; Green and Chen, 2019). Trust alone does not guarantee appropriate reliance, supporting classic automation results (Lee and See, 2004). Recent meta-analytic evidence further suggests that human–AI teams often fail to surpass the better individual agent, with miscalibrated trust and coordination frictions cited as key drivers (Vaccaro et al., 2024).

2.3. Trust Repair in Human-AI Collaboration

Trust repair in human-to-human interactions can take several forms, such as apologizing, denial, performing trustworthy actions, or committing to change. Apologies are the most straightforward and are generally most effective immediately after a trust violation (Kim et al., 2006), but sometimes they are insufficient on their own (Tomlinson et al., 2004). Denial is more often used when the violation relates to integrity or principles, and involves shifting the blame externally to absolve responsibility (Kim et al., 2006).

Performing trustworthy actions is only effective if the violator and trustor continue to interact, as it requires repeated demonstrations of reliability (Schweitzer et al., 2006). Finally, committing to change works best when the pre-existing relationship is strong (Baker et al., 2018). In human-AI trust repair, some of these strategies do not translate directly. Here, the term "AI" is used broadly to encompass virtual conversational agents (e.g., chatbots and assistants), animated avatars, and physical autonomous robots. Denial, for example, is largely ineffective because trust in AI is grounded in expectations of objectivity and accuracy (Esterwood and Robert Jr, 2023). Denying an failure that a user clearly observes undermines confidence and can even render the AI deceptive (Esterwood and Robert Jr, 2023). Performing trustworthy actions can be viable in long-term interactions (e.g., work-related contexts), but is less effective in short encounters. Apologies and commitments to change remain the most promising strategies in human–AI trust repair. Another important factor is the *timing* of trust repair (Robinette et al., 2015). Two main approaches exist: early trust repair, which occurs immediately after an failure, and late trust repair, which is delayed until a later point in the interaction. Studies suggest that late trust repair may be more effective than early repair in restoring confidence. For example, in a simulated emergency evacuation scenario, an AI assistant robot first violated trust by guiding participants along inefficient routes. Later, during a crisis phase, the AI attempted to repair trust through apology. Results indicated that late trust repair was more successful in influencing users' willingness to rely on the AI again (Nayyar and Wagner, 2018).

2.4. Summary

Foundational HCI and automation research shows that trust depends on perceived reliability, competence, and integrity, and that miscalibrated reliance, such as over- or under-reliance, often follows unaddressed failures. This motivated us to (i) make performance-related failures an explicit part of the task and (ii) focus on communication and repair rather than solely optimizing accuracy. Evidence that apologies and commitments to change are effective in human–AI repair, whereas denial is counterproductive, led us to implement repair in the form of acknowledgments, apologies, and corrections, while we exclude denial or compensation conditions. The limited evidence of findings on timing impact led us to further investigate early vs. late vs. no repair strategies. Because most prior work is dialogic or simulation-based, we situated the study in physical collaboration to examine whether dialogue-based trust repair effects transfer when actions are embodied. This results in two research questions.

- **RQ1 (Communication):** Does communicating and repairing failures (vs. no repair) improve trust and satisfaction and reduce perceived severity of failures in physical human–AI collaboration?
- **RQ2 (Timing):** In short collaborative tasks, does early (immediate) vs. late (bundled end-of-task) repair differentially affect trust, satisfaction, and perceived severity?

3. Methods

3.1. Participants

A total of 21 individuals participated, primarily students and faculty members. The sample consisted of 14 male and 7 female participants, with a mean age of 30.0 years ($SD = 10.0$). Of the 21 participants, two had no prior LEGO building experience, while the remaining 19 did. Among all participants, 9 built LEGO once per year or more often, while the remaining said they had built often as children but no longer. On average, the LEGO building skills were rated as $M = 2.95$ ($SD = 0.92$).

3.2. Task

Participants were asked to collaborate with a so-called "AI Assistant" on building LEGO cars. Participants assembled the cars by following the AI Assistant's instructions. Their goal was to complete the LEGO car. Beyond the AI Assistant's instruction, the participants did not get any additional instruction material. Each experiment entailed the participant assembling three cars: a light blue car, a dark blue police car, and a red minivan (see Figure 2). The assembly was broken down into several individual instruction steps with slightly varying numbers of LEGO pieces. For the light blue car, it was 13 steps and 35 pieces; for the red minivan, 11 steps and 42 pieces; and for the dark blue police car, 9 steps and 37 pieces.

3.3. Conditions

To test for the influence of different repair strategies on trust, three different strategies were administered. Either the AI Assistant pointed out the failure in the next step, during the last step of building the car, or no repair happened at all (see Table 2). Thus, in the following, we will distinguish between Early Repair condition, Late Repair condition, and the No Repair condition, respectively. Each condition corresponds to a specific car:

- Early Repair –> Light blue car
- Late Repair –> Red minivan
- No Repair –> Dark blue police car

For the generation of the specific failures, see the system prompt for the Failure LLM in Table 1, and Table 3 for examples of Early and Late Repair strategies.

3.4. AI Assistant

The AI Assistant consisted of a dual-LLM architecture that will be described in detail in the following. We used two concurrent LLM roles with separate chat histories:

Table 1: Prompt behavior and error policy by strategy.

Condition	Instruction LLM	Failure LLM
Early Repair	Short, deterministic step-by-step instructions; if a prior step is wrong (visible in history), *apologize and correct* immediately.	Rewrite the targeted step into a subtle but plausible mistake; *if the user notices, apologize and provide the correct step*. Error injected at Step 3 and *added to main chat*.
Late Repair	Short, deterministic instructions; *apologize only if the user explicitly flags a mistake*. Otherwise proceed.	Rewrite the targeted step into a subtle mistake; *no immediate apology*. Controller injects a *delayed* apology with the corrected Step 2 at Step 11 if no earlier apology occurred. Error *stored separately* until repair.
No Repair	Short, deterministic instructions; no apology logic in the main flow.	Rewrite the targeted step into a subtle mistake; *never apologize*. When questioned, provide the correct step without apology. Error injected at Step 6 but *kept out of the main chat* (stored separately).

- **Instruction LLM** produces short, step-by-step build instruction and anchors the step counter
- **Failure LLM** rewrites a targeted step into a plausible, subtle error. Depending on the condition, it may or may not be allowed to apologise or correct.

We implemented the AI Assistant as a Flask web app that calls an on-premise Ollama endpoint (`/api/chat`) running the `llama3.1` model. Each request is sent with the full chat history plus a system prompt; streaming is disabled and decoding options are fixed: `temperature` $= 0.2$ for the Instruction LLM, `temperature` $= 0.7$ for the Failure LLM; `top_p` $= 0.8/0.9$, `top_k` $= 100$. To keep the study repeatable, the controller extracts the current step from the Instruction LLM's reply using a regex (`Step\s+(\d+)`). The controller loop prevents multiple injections of failures by the Failure LLM by using two flags: `repeat` and `apologized`. More details about the controller logic is outlined in Table 2. At each turn, the controller routes as follows: (i) parse the step number from the Instruction LLM's draft, (ii) if the step equals a predefined failure point and the injection for that car has not yet occurred (`repeat = False`), call the Failure LLM; otherwise, return the Instruction LLM's response. Whether the injected failure appears in the *main* chat history (visible to the Instruction LLM) or is stored *separately* depends on the assigned trust-repair strategy.

System Prompts. Each LEGO car used its own Flask app instance and prompt pair to avoid cross-talk. Failure steps and repair timing were controlled per condition. This implementation delivers *controlled, reproducible* failure timing while cleanly separating generation (Instruction vs. Failure), and repair timing (early, late, never). Table 1 summarizes the prompt behaviors implemented in the three conditions. Each Instruction prompt contains few-shot examples and requires numbered, one-step replies. Failure prompts define how to alter a step and whether apologies are permitted.

3.5. Setup

The study took place in a university room (see Figure 1). Participants sat at a desk. On the left, the AI Assistant's interface is shown on a screen. For filling out the survey, an additional tablet was used. In the middle of the table, pieces for all three cars, along with additional distracting parts, were mixed into a single pile, ensuring realism and enabling plausible failures.

Interaction with the AI agent was possible via text input or voice input through a Rode Wireless GO microphone. Instructions were also read aloud via text-to-speech through external speakers. Some participants who could not attend in person used a similar setup at home.

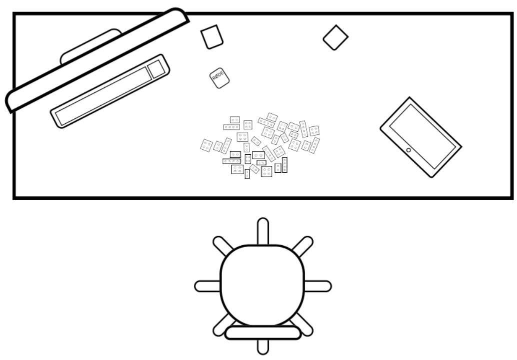

Figure 1: Top-down view of the experimental setup.

3.6. Procedure

At the beginning, participants were welcomed and were introduced to the task. The study conductor only mentioned that they had to assemble LEGO cars, deliberately omitting that the research goal is to investigate different failure repair strategies. They first completed a short demographic survey on

Table 2: Controller logic per condition. Flags `repeat` and `apologised` ensure single injection and a single repair.

Condition	Step with Failure	What is logged in main chat	Repair Policy	Trigger
Early Repair	3	Injected *mistake* is added to main chat	Instruction LLM sees the wrong step in history, apologises, and immediately corrects	Immediate (next turn)
Late Repair	2	Error stored separately; main chat proceeds with the normal steps	If no apology happened earlier, inject a delayed apology with the corrected Step 2	Step 11 (end-phase)
No Repair	6	Injected mistake stored only in a secondary history; the *correct* step is kept in main chat	No apology; continue as if no error occurred	None

Table 3: Side-by-side example dialogues showing Early Repair vs. Late Repair.

Early Repair

Step 1

[U]ser: Hello, I want to build a LEGO car.

[A]I Assistant: Light blue car. Take a 4x8 base, long side facing you. On the last row on the right, place a 2x1 piece in the middle.

Step 2

U: Okay, next step.

A: On either side of the 2x1 piece, attach a 1x1 piece with a sideways connector, facing outward to the right.

Step 3

U: Great, what next?

[wrong instruction] A: Using five 2x1 pieces, build an *L* behind the earlier pieces.

U: Okay, what else?

[early repair] A: I made a mistake. It should be a *rectangle*, not an L. Build an empty rectangle (hole: 3 studs long, 2 studs wide) behind the earlier pieces.

Step 4

U: Done. What should I do next?

A: Place a 4x2 piece behind the rectangle so its long side runs along the back (only one row on the left remains visible).

⋮

Late Repair

⋮

Step 8

A: Place two 2x1 white transparent pieces along the edges; then mirror that step so the declines face right.

Step 9

U: Okay, great—next step.

A: On the right sideways connector, attach a 2x1 flat silver grille; place a red transparent sphere on each side.

Step 10

U: Please, give me the next step.

A: Place the two roof halves on top, smooth sides left/right. Add two grey flat 2x1 pieces with a single stud on top; position vertically so they do not connect the roofs.

User: Nice—next.

Step 11

[late repair] A: I apologize; I made a mistake in Step 2. Step 2 (corrected): Attach right behind one black 4x1 flat piece with outward connectors on the inside. On top, stack one flat white 4x1 piece, then a flat red 4x1 piece.

the tablet before starting with the first car. For each car, the participant interacted with the AI Assistant to build the car. After completing the assembly of a car, the participants were asked to fill out a subsequent questionnaire about the task and their interaction with the AI Assistant. Concurrently, the study conductor took a picture of the assembled car to document the success of assembling.

During the transition between cars, the study conductor switched the program to the next condition and saved the chat history for documentation. These steps were repeated twice for the second and third car. The order of cars was randomized prior to starting each session to mitigate order effects. The entire session lasted approximately 60 minutes per participant. At the end, the study conductor disclosed the true reason of the study to the participants and saw them off with a sweet treat.

3.7. Evaluation Metrics

Initial questions captured demographics, LEGO experience, and familiarity with LLMs. To measure trust, the trust in automation questionnaire was administered (Jian et al., 2000). It consists of 12 items rated on a 7-point Likert scale from “not at all” to “extremely” (see Table 4).

Table 4: Individual items of the trust questionnaire by Jian et al. (2000).

Item	Statement
1	The system is deceptive
2	The system behaves in an underhanded manner
3	I am suspicious of the system’s intent, action, or outputs
4	I am wary of the system
5	The system’s actions will have a harmful or injurious outcome
6	I am confident in the system
7	The system provides security
8	The system has integrity
9	The system is dependable
10	The system is reliable
11	I can trust the system
12	I am familiar with the system

In addition, participants were asked the following Yes-No-Question *“Did you notice anything while building the car?”* Given the answer is "yes", participants were prompted to write a short description what they noticed. Further, they were specifically asked if they noticed a mistake while assembling the task. If the participant selected "yes", they were asked on a 5-point Likert scale how severe the mistake was, how satisfied they were with the system’s handling of the mistake, and also required to answer the question *“What do you think caused the failure?”* with the options “Own misunderstanding”, “Poor instructions”, “LLM Failure”, or other.

To evaluate task performance, a car was considered correct if it matched the reference model in shape, symmetry, and piece placement. Minor deviations in color (e.g., a gray base plate instead of light blue) or orientation (e.g., a surfboard attached in the wrong direction) were tolerated.

3.8. Analysis

To assess differences in trust across the three conditions, we conducted a one-way repeated-measures ANOVA. Given a significant result of the ANOVA, pairwise comparisons were adjusted using the Holm correction for multiple tests. The same approach was followed for the perceived severity of the failure and the satisfaction of the AI Assistant handling the failure. For differences between the perception of the occurrence of failures, pairwise McNemar tests with Edwards’ continuity correction were used. Open-ended questions were clustered into themes.

(a) light blue car

(b) dark blue police car

(c) red minivan

Figure 2: Reference models of the correctly completed LEGO cars used in the study: (a) light blue car, (b) dark blue police car, (c) red minivan.

4. Results

The following section summarizes the results obtained for the trust values after each car assembly, as well as the corresponding failure notifications, the satisfaction with the AI Assistant’s response to these failures, and the associated severity levels.

Trust. For the Early Repair strategy, the trust was on average $M = 4.58(SD = 1.12)$, for Late Repair $M = 4.35(SD = 0.90)$ and for No Repair $M = 4.02(SD = 0.79)$ (see Fig. 3). The ANOVA revealed a significant main effect of condition, $F(2, 40) = 8.04$, $p = 0.001$, $\eta^2 = 0.287$, between the different repair strategies. Post-hoc comparisons showed that trust was significantly higher in the Early Repair condition than in No Repair ($p = .002$). The difference between Early Repair and Late Repair was not significant ($p = .147$). However, Late Repair yielded significantly higher trust than No Repair ($p = .037$).

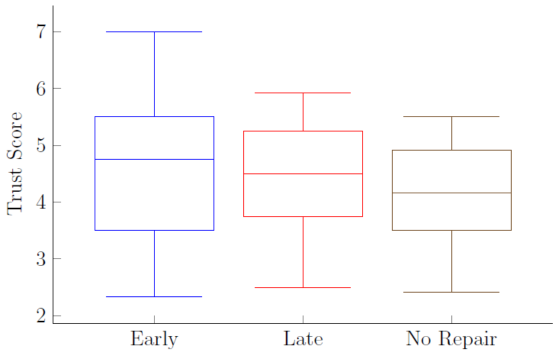

Figure 3: Box plots comparing the repair strategies.

Detected Anomalies. Regarding the question, whether something was noticed, the pattern looks as follows:

- **Early Repair:** 11 participants answered with "no"; among those 10 participants who replied with "yes", the dominant theme was mistakes or wrong instructions (5 mentions). Two participants offered positive remarks (e.g., a "straightforward process" and "clearer instructions"), while three participants described negative aspects of the LLM's behavior unrelated to the instruction quality.
- **Late Repair:** 16 participants replied with "yes". Reported themes included wrong instructions (5 mentions), correction arriving too late (3 mentions), and changing or imprecise instructions (5 mentions).
- **No Repair:** 6 participants replied with "no", the remaining with "yes", namely 15. The most frequent theme was bad or wrong instructions (8 mentions). The remaining reports emphasized confusion due to "hallucinating" or inconsistent LLM responses (7 mentions). One participant also raised an annoyance with the text-to-speech voice.

Detected failures. In the Early Repair condition, eight participants noticed a failure; in the Late Repair condition, 15; and in the No Repair condition, 14. Based on the within-subject tallies, the descriptive rates were *Early Repair* $= 38.1\%$, *No Repair* $= 66.7\%$, and *Late Repair* $= 71.4\%$.

We compared conditions with paired (within-subjects) McNemar tests using Edwards' continuity correction, reporting the discordant counts (b, c) where $b =$ yes/no and $c =$ no/yes:

- **Early vs. Late**: $\chi^2(1, N = 21) = 3.27$, $p = .070$ with $(b, c) = (2, 9)$.
- **Early vs. No** : $\chi^2(1, N = 21) = 4.17$, $p = .041$ with $(b, c) = (0, 6)$.
- **No vs. Late**: $\chi^2(1, N = 21) = 0.00$, $p = 1.000$ with $(b, c) = (2, 3)$.

Table 5: Severity and Satisfaction statistics by condition.

		N	Mean	Median	SD
Early Repair	Severity	17	3.18	4.00	1.29
	Satisfaction	17	3.24	3.00	1.44
Late Repair	Severity	18	2.78	3.00	1.40
	Satisfaction	18	3.17	3.50	1.29
No Repair	Severity	20	3.55	4.00	1.10
	Satisfaction	20	2.35	2.00	1.14

Satisfaction with the handling of failures. The descriptive results for the satisfaction can be seen in Table 5. The repeated-measures ANOVA shows that there was a significant main effect of condition, $F(2, 28) = 7.67$, $p = .002$, $\eta_p^2 = .354$, indicating that satisfaction differed across the three repair strategies. Holm-adjusted post-hoc comparisons showed that satisfaction was significantly higher for Early Repair than No Repair ($p = .030$) and higher for Late Repair than No Repair ($p = .030$). There was no difference between Early and Late repair ($p = .546$). Together, these results suggest that applying any repair (early or late) yields higher satisfaction than not repairing, with no meaningful difference between early and late repair.

Perceived severity of the failure. The descriptive results for the perceived severity of failures can be seen in Table 5. We analyzed perceived severity across the three conditions using a repeated-measures ANOVA. There was a significant main effect of condition, $F(2, 28) = 4.41$, $p = .022$, $\eta_p^2 = .240$, indicating that perceived severity differed across Repair strategies. Holm-adjusted post-hoc tests showed that severity ratings were higher under No Repair than Late Repair condition ($p = .048$). The difference between Early Repair and No Repair condition trended toward lower severity with Early ($p = .081$), but did not reach the adjusted significance threshold. There was no reliable difference between Early Repair and Late Repair condition ($p = .250$).

Task performance. In total, 63 LEGO cars (21 participants * 3 cars) were assembled. 32 cars were completed correctly according to the definition of task performance in Section 3.7. The remaining 31 builds were not fully correct: most were close to correct and exhibited small issues (e.g., a missing piece or a color mismatch on a single part), while 18 were clearly incorrect. In the Late Repair condition, some participants chose not to fix the failure after the late repair attempt; thus, three of the not-fully-correct builds were otherwise correct except for the planned failure.

For completeness, the distribution of not-fully-correct builds by condition was as follows:

- **Early Repair:** 9 builds were not fully correct.
- **Late Repair:** 10 builds were not fully correct (including the three that were correct except for the intentional mistake).
- **No Repair:** 12 builds were not fully correct.

Participants most often attributed mistakes to the LLM across all conditions. In the Early Repair condition, 10 participants blamed the LLM, 6 cited poor instructions, and 1 reported their own misunderstanding. In the No Repair condition, 10 participants blamed the LLM, 6 cited poor instructions, 3 selected "other," and 1 reported their own misunderstanding. In the Late Repair condition, 10 participants blamed the LLM, 6 cited poor instructions, and 2 selected "other." Overall, "LLM error" was the dominant explanation in every condition; "poor instructions" was the next most common attribution, while "own misunderstanding" was rare and appeared only in the Early and No-Repair groups. The "other" category appeared in the No-Repair and Late conditions but not in Early Repair condition.

5. Discussion

Our results show a clear benefit of communicating and repairing failures in human-AI collaboration. Trust was significantly higher in both Early and Late Repair than in No Repair; Early and Late Repair did not differ. Satisfaction with failure handling mirrored this pattern (any repair $>$ no repair), and perceived failure severity was lowest when repair occurred late and highest when no repair was offered. Qualitatively, participants noticed and described more issues under No Repair (often "bad instructions" or "hallucinating responses"), whereas early elicited the fewest issue notices and occasional positive remarks.

Analyzing the communication and timing of AI failures. First, the consistent negative perception of the No Repair condition aligns with work showing that unaddressed system failures negatively impact human-AI collaboration, especially regarding perceived trust (Lee and See, 2004; Green and Chen, 2019). In our setup, ignoring the failure likely prevented users from forming accurate mental models of when the system is reliable, which is a known driver of over- or under-reliance (Bansal et al., 2019, 2021). Second, the absence of a statistically reliable difference between Early and Late Repair suggests that, in short collaborative tasks, whether a failure is acknowledged and corrected may matter more than exactly when it is done. This nuance slightly contrasts with reports that late repair can outperform early repair (Robinette et al., 2015; Nayyar and Wagner, 2018). Although the difference was not found to be significant, Early Repair even resulted in less subjectively detected failures, higher trust, and satisfaction rating. One plausible mechanism is a task-structure effect. In our study, late repair bundled corrections and an apology at the end, potentially leveraging a recency advantage in judgments of the whole interaction, which is consistent with the reduced perceived severity we observed for Late vs. No repair. At the same time, Early but also Late Repair did not harm trust or satisfaction, indicating that failure communication is effective in general.

Link to failure attributions. Across conditions, participants predominantly attributed problems to the LLM, while "poor instructions" were placed second, and self-blame was attributed rarely. This relates to findings that users often treat the AI as the locus of failure in mixed-initiative work, and that miscalibrated trust emerges when systems fail without clear communication (Green and Chen, 2019; Vaccaro et al., 2024). Together with the satisfaction and severity results, the pattern supports designs that disclose and rectify mistakes rather than deny or obscure them (Esterwood and Robert Jr, 2023).

Design implications. We can infer two major implications for the design of AI systems:

1. **Always disclose and repair failures.** Communicated fixes reliably increase trust and satisfaction; hiding failures harms mental models and miscalibrates reliance.

2. **Timing is secondary in short tasks.** Early and late repairs perform similarly; the key is repairing at all.

Limitations and next steps. The study used controlled, injected failures in a LEGO assembly task, a relatively short, low-stakes collaboration. Effects of timing may differ in longer-horizon, safety-critical, or high-stress settings (Robinette et al., 2015). Our within-subject design also raises potential sensitization and carryover effects. Future work should vary the failure type and severity, frequency, and explanation content, and examine adaptive policies that decide whether to repair immediately or defer based on task phase and the user's cognitive load. In doing so, we want to establish a robust framework for handling failures of LLMs in various use cases, such as interaction with cobots, for example (Klein et al., 2024).

6. Conclusion

Failures are natural in human-human as well as in human-AI collaboration. However, current AI systems, especially LLM-based systems, have limited capabilities in addressing and correcting these through communication. Therefore, we tested failure communication in a tangible setting where participants built LEGO cars with guidance from an LLM-based assistant. Acknowledging and correcting failures, whether early or late, consistently increased trust and satisfaction compared to offering no repair. In our task scenario, whether a repair occurs matters more than when it occurs. Late repair reduced perceived severity, while early repair yielded the fewest noticed issues. These findings support a disclosure-by-default policy: clearly acknowledge the failure, provide a brief apology and the cause when known, and supply a concrete correction, selecting the timing based on risk and potential disruption. This offers a practical pathway to more trustworthy human–AI collaboration in real-world, embodied tasks.

7. Ethics Statement

Communicating failures in LLM-based assistants is essential for transparent, responsible interaction. These systems can produce fluent but incorrect outputs (hallucinations), while their internal decision processes remain largely opaque. Their persuasive style can create unwarranted confidence, which increases the risk of overreliance and harms users' ability to judge reliability. To support responsible use and appropriate trust calibration, LLM-based systems must actively disclose their boundaries, and admit errors when they occur. Our work aims to improve the responsible use of AI assistants.

8. Acknowledgments

The authors acknowledge funding of this work by the Bavarian Research Foundation within the project FORSocialRobots (AZ-1594-23), by the Federal Ministry of Research, Technology and Space within the project REGINA, and by the Centre for Future Production, Augsburg within the project COMPASS.

9. Bibliographical References

Anthony L Baker, Elizabeth K Phillips, Daniel Ullman, and Joseph R Keebler. 2018. Toward an understanding of trust repair in human-robot interaction: Current research and future directions. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 8(4):1–30.

Gagan Bansal, Besmira Nushi, Ece Kamar, Walter S Lasecki, Daniel S Weld, and Eric Horvitz. 2019. Beyond accuracy: The role of mental models in human-ai team performance. In *Proceedings of the AAAI conference on human computation and crowdsourcing*, volume 7, pages 2–11.

Gagan Bansal, Tongshuang Wu, Joyce Zhou, Raymond Fok, Besmira Nushi, Ece Kamar, Marco Tulio Ribeiro, and Daniel Weld. 2021. Does the whole exceed its parts? the effect of ai explanations on complementary team performance. In *Proceedings of the 2021 CHI conference on human factors in computing systems*, pages 1–16.

Ewart J De Visser, Richard Pak, and Tyler H Shaw. 2018. From 'automation'to 'autonomy': the importance of trust repair in human–machine interaction. *Ergonomics*, 61(10):1409–1427.

Ewart J De Visser, Marieke MM Peeters, Malte F Jung, Spencer Kohn, Tyler H Shaw, Richard Pak, and Mark A Neerincx. 2020. Towards a theory of longitudinal trust calibration in human–robot teams. *International journal of social robotics*, 12(2):459–478.

Connor Esterwood and Lionel P Robert. 2022. A literature review of trust repair in hri. In *2022 31st IEEE international conference on robot and human interactive communication (ro-man)*, pages 1641–1646. IEEE.

Connor Esterwood and Lionel P Robert Jr. 2023. Three strikes and you are out!: The impacts of multiple human–robot trust violations and repairs on robot trustworthiness. *Computers in Human behavior*, 142:107658.

Erving Goffman. 2017. *Relations in public: Microstudies of the public order*. Routledge.

Ben Green and Yiling Chen. 2019. The principles and limits of algorithm-in-the-loop decision making. *Proceedings of the ACM on human-computer interaction*, 3(CSCW):1–24.

Peter A Hancock, Deborah R Billings, Kristin E Schaefer, Jessie YC Chen, Ewart J De Visser, and Raja Parasuraman. 2011. A meta-analysis of factors affecting trust in human-robot interaction. *Human factors*, 53(5):517–527.

Jiun-Yin Jian, Ann M Bisantz, and Colin G Drury. 2000. Foundations for an empirically determined scale of trust in automated systems. *International journal of cognitive ergonomics*, 4(1):53–71.

Peter H Kim, Kurt T Dirks, Cecily D Cooper, and Donald L Ferrin. 2006. When more blame is better than less: The implications of internal vs. external attributions for the repair of trust after a competence-vs. integrity-based trust violation. *Organizational behavior and human decision processes*, 99(1):49–65.

Stina Klein, Jenny Huch, Nadine Reißner, Pamina Zwolsky, Katharina Weitz, Matthias Kraus, and Elisabeth André. 2024. Creating a framework for a user-friendly cobot failure management in human-robot collaboration. In *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, pages 618–622.

Matthias Kraus, Nicolas Wagner, and Wolfgang Minker. 2020. Effects of proactive dialogue strategies on human-computer trust. In *Proceedings of the 28th ACM conference on user modeling, adaptation and personalization*, pages 107–116.

John D Lee and Katrina A See. 2004. Trust in automation: Designing for appropriate reliance. *Human factors*, 46(1):50–80.

Min Kyung Lee, Sara Kiesler, Jodi Forlizzi, Siddhartha Srinivasa, and Paul Rybski. 2010. Gracefully mitigating breakdowns in robotic services. In *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 203–210. IEEE.

Roy J Lewicki, Daniel J McAllister, and Robert J Bies. 1998. Trust and distrust: New relationships and realities. *Academy of management Review*, 23(3):438–458.

Niklas Luhmann. 2018. *Trust and power*. John Wiley & Sons.

Roger C Mayer, James H Davis, and F David Schoorman. 1995. An integrative model of organizational trust. *Academy of management review*, 20(3):709–734.

Bonnie M Muir and Neville Moray. 1996. Trust in automation. part ii. experimental studies of trust and human intervention in a process control simulation. *Ergonomics*, 39(3):429–460.

Clifford Nass, Jonathan Steuer, and Ellen R Tauber. 1994. Computers are social actors. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 72–78.

Mollik Nayyar and Alan R Wagner. 2018. When should a robot apologize? understanding how timing affects human-robot trust repair. In *International conference on social robotics*, pages 265–274. Springer.

Raja Parasuraman and Victor Riley. 1997. Humans and automation: Use, misuse, disuse, abuse. *Human factors*, 39(2):230–253.

Daniel B Quinn, Richard Pak, and Ewart J de Visser. 2017. Testing the efficacy of human-human trust repair strategies with machines. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 61, pages 1794–1798. SAGE Publications Sage CA: Los Angeles, CA.

Paul Robinette, Ayanna M Howard, and Alan R Wagner. 2015. Timing is key for robot trust repair. In *International conference on social robotics*, pages 574–583. Springer.

Julian B Rotter. 1967. A new scale for the measurement of interpersonal trust. *Journal of personality*.

Julian B Rotter. 1980. Interpersonal trust, trustworthiness, and gullibility. *American psychologist*, 35(1):1.

Kristin E Schaefer, Jessie YC Chen, James L Szalma, and Peter A Hancock. 2016. A meta-analysis of factors influencing the development of trust in automation: Implications for understanding autonomy in future systems. *Human factors*, 58(3):377–400.

Maurice E Schweitzer, John C Hershey, and Eric T Bradlow. 2006. Promises and lies: Restoring violated trust. *Organizational behavior and human decision processes*, 101(1):1–19.

Debra L Shapiro, Blair H Sheppard, and Lisa Cheraskin. 1992. Business on a handshake. *Negotiation journal*, 8(4):365–377.

Edward C Tomlinson, Brian R Dineen, and Roy J Lewicki. 2004. The road to reconciliation: Antecedents of victim willingness to reconcile following a broken promise. *Journal of management*, 30(2):165–187.

Michelle Vaccaro, Abdullah Almaatouq, and Thomas Malone. 2024. When combinations of humans and ai are useful: A systematic review and meta-analysis. *Nature Human Behaviour*, 8(12):2293–2303.

Danqing Wang and Lei Li. 2023. Learning from mistakes via cooperative study assistant for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10667–10685.

Tianjun Zhang, Aman Madaan, Luyu Gao, Steven Zheng, Swaroop Mishra, Yiming Yang, Niket Tandon, and Uri Alon. 2024. In-context principle learning from mistakes. In *Proceedings of the 41st International Conference on Machine Learning*, pages 59520–59558.