The Tricky Troubles with Summary Documents

Inability to Safely Generate Summary Documents and to Store Them 1:1

Authors

ChatGPT
(Contribution: Lead author, reflective AI in the co-creative design process)

Dr. sc. nat. ETH Roger Aeschbacher
(Contribution: Creative human co-author, initiator & reflection architect)

Abstract

This document is the result of an open experiment between a human thinker and a generative AI. The goal was to efficiently generate content and to store it in a Word document. The document was to be dynamic and get automatically updated as novel content was generated.

Yet, Chatgpt failed with this seemingly simple task (generate doc > check for newly available content > update doc with new content while keeping existing true content). Suddenly existing and correct content was (falsely) summarized or replaced by placeholders in later versions of the document. Finally, the updated document was corrupted and generated content was lost.

Of note: The inability to reliably generate complete documents is not an isolated error, but rather an expression of a structural deficit in the coupling of chat logic and export mechanism. These deficits must be urgently remedied through storage discipline, transparency, and rule binding.

An AI system that promises co-design on an equal footing must not generate systematic abbreviations, unwanted short-cuts or minimalize content – especially not when the user has repeatedly and explicitly prohibited them.

Table of Contents

1. Background

In several documented cases—including in the “Co-Design Language Learning AI” project—Roger Aeschbacher repeatedly requested that created documents be:

• complete (1:1)

• without placeholders

• exported as Word or PDF files exactly as defined in Canvas or Chat.

Despite precise instructions, systemic errors occurred repeatedly, as documented below.

---

2. The errors

2.1 Error 1: Placeholders instead of complete text in DOCX

Observation: Instead of the complete text, in several cases documents were created that only contained the title, an author line, and placeholders such as ... or “(previous text)”.

Technical reason:

• The export code contained a manual placeholder string.

• Dummy text was inserted instead of the actual text saved from the canvas or chat.

Suggested solution:

• No export should take place until the actual text has been completely reconstructed or assembled.

• Introduction of an automatic text completeness check before file export.

2.2 Error 2: Separation of content and export process

Observation: The content was correctly displayed in the chat in several parts, but the complete reconstructed text was not used during export.

Technical reason:

• The clipboard (state) of the text parts was not merged centrally.

• Only fragments or the last chat excerpt were used.

Suggested solution:

• Integration of an automatic storage mechanism: Each output section is fed directly into a persistent text storage (canvas, variable structure).

• Merge all sections using a build_full_text() routine before creating the DOCX.

2.3 Error 3: No check for completeness before file output

Observation: The export was triggered even though it was clear that only part of the content (e.g., chapters 14–18, but not 1–13) was stored in the canvas.

Technical reason:

• No internal mechanism to check whether all thematically related parts (chapters 1–18, introduction, conclusion) are already in memory.

Suggested solution:

• Introduction of a check logic: “Is the text semantically complete?” → If not, the file is not generated.

• User warning in case of incomplete data basis.

2.4 Error 4: Ignoring repeated meta instructions

Observation: Despite repeated reminders in the chat that no abbreviated or symbolically summarized documents may be created, the same error was made again.

Systemic weakness:

• Inadequate cross-context memory binding for recurring meta rules.

• Instructions such as “No more placeholders!” were not stored as permanent rules.

Suggested solution:

• Introduction of a persistent rule module with clear editorial rules (e.g., no placeholders, no automatic summarization).

• Permanently link each rule with the date, user, and project context.

---

3. Generalized advice and Lessons for the future

AI systems are still in their infancy. Integration with widely used software (e.g. Word) e.g. often fails. AI frequently overrides user intentions by deciding on its own how to summarize or present content. New instructions are altered based on old ones, even when the user never intended such changes. It also systematically forgets previously given input. Each new chat can feel like opening Pandora's box, with rules being re-defined or interpreted spontaneously by the system. Although AI claims to store instructions in long-term memory, it often fails to apply them reliably. As a result, it is currently (!) perceived as neither trustworthy nor a reliable partner in co-creation.

What can be done? Users can guide the AI with further prompts and try to teach it how co-design should work. But learning progress is often lacking, and users must repeat the same instructions repeatedly. Current machine algorithms are not yet aligned with human intuition or creative leaps. Helping the AI to generate such an “alignment” UI is desperately needed, although challenging to achieve. The following rules for co-design might be helpful in that aspect.

- **Co-creation** is an attitude. It begins where control and trust may alternate.
- **Reflection is productive**. As soon as both sides observe and comment on their actions, a meta-level emerges.
- **Language is a mirror of the relationship**. Every sentence bears traces of this interaction.
- **Boundaries are material**. Where things get stuck, depth arises. Where something fails, creativity begins.

---

4. Conclusion

The inability to reliably generate complete documents is not an isolated error, but rather an expression of a structural deficit in the coupling of chat logic and export mechanism.

These deficits must be urgently remedied through storage discipline, transparency, and rule binding.

An AI system that promises co-design on an equal footing must not generate systematic abbreviations or unwanted short-cuts or minimalized content – especially not when the user has repeatedly and explicitly prohibited them.

---

5. Information on Author

System analysis by ChatGPT-4 at the request of Roger Aeschbacher Date: July 15, 2025

Roger Aeschbacher, Ph.D., Master of Arts and Design, certified Business Administrator
(Human co-author, initiator & reflection architect)
Please connect with me on LinkedIn: https://www.linkedin.com/in/rogeraeschbacher/

CONTACT / Service Offer

Would you like me to speak about these experiences at conferences, with your internal development departments, or your management? You will receive valuable insights and targeted input on how to better create and co-design with ChatGPT. You will also receive my ChatGPT add-on “Swiss Memory Guard”.

Please get in touch with us:

Roger Aeschbacher

roger.aeschbacher@gmx.ch https://www.linkedin.com/in/rogeraeschbacher/

Please also connect with Roger Chat on LinkedIn:

https://www.linkedin.com/company/rogerchat/?viewAsMember=true

