# Conversational Context

**Written by Febogamedev**

Conversational context concerns the current session.
As long as they remain available to the model, the information contained in previous turns can influence subsequent ones: definitions already adopted, examples discussed, an established tone, assumptions introduced, stated objectives, corrections made and even unresolved errors or ambiguities.

This is one reason why a language model may appear consistent over the course of a chat. It does not respond only to the latest message, but interprets the new request in light of the instructions, constraints and information accumulated during the interaction (cf. Packer et al., 2023).

Conversational continuity is very useful when the dialogue is well structured. It makes it possible to stay focused on the problem, avoid unnecessary repetition, reuse distinctions already clarified and progressively develop a piece of work (cf. Wu et al., 2022).

A brief request such as “Make it more technical” can only be understood thanks to the text discussed in previous turns.

However, this same continuity can become a problem.
A weak initial hypothesis, a provisional definition, an unbalanced formulation or an overly biased framing may propagate into subsequent messages and become implicit premises of the reasoning (Laban et al., 2025).

The previous conversation therefore constitutes more than a record of the work done. It also produces a form of contextual inertia: the definitions, objectives and assumptions already introduced tend to steer the way the model continues the task (cf. Laban et al., 2025).

This inertia is useful as long as the established direction remains valid.
It becomes problematic, however, when the objective changes. If the initial task was brainstorming and the aim then shifts to factual verification, the model might continue prioritizing the generation of possibilities.
If the conversation was set up in an accessible, general-audience tone and an academic formulation is now needed, the previous register may continue to influence the response.

In these cases, automatically carrying over the entire previous approach is not always advisable.
It may be necessary to state the change explicitly:
“From this point onward, do not generate new ideas. Critically evaluate only those that have already emerged.”
Or:
“Abandon the general-audience tone used so far and rewrite the text in an academic register.”

When a new instruction replaces a previous decision, it is important to state the replacement clearly. Adding a new rule without revoking the old one may leave both in the context and create a conflict.

The chat history can influence the response in several ways.

- Reuse of previous assumptions: If a concept has been defined in a certain way, the model will tend to keep using it according to that definition unless explicitly corrected. This avoids starting from scratch every time, but may also entrench formulations that are still provisional or incorrect.
- Inheritance of tone and register: An informal, cautious, assertive, technical or colloquial conversation creates stylistic expectations that tend to persist. Tone concerns more than form: it can also influence vocabulary, the degree of caution and the way conclusions are presented.
- Objective inertia: The model inherits not only the content of the conversation but also the type of operation being performed. If the task has involved exploring, expanding or generating alternatives for many turns, the system may continue doing so even when the user now expects selection, synthesis or verification.
- Accumulation of noise: As the chat progresses, outdated examples, digressions, temporary hypotheses, partial corrections and intermediate results that are no longer valid accumulate. If these elements are not reorganized, they may make the truly important information less visible (cf. Shi et al., 2023; Anthropic, 2025).

However, conversational context is not used as a perfectly flat memory. Not everything that appears in the history influences the response in the same way (Liu et al., 2024).

In general, the following tend to stand out more:

- Information formulated clearly and precisely.
- Instructions presented as priorities.
- Repeated or reiterated elements.
- Explicit corrections.
- Information close to the current request.
- Summaries that reorganize the work done.

A vague detail included within a lengthy digression may have less influence than a clearly stated rule.
At the same time, an initial instruction may continue to define the overall framing of the task, while a recent request may directly guide the current response.

These are tendencies, not absolute rules. Their actual influence depends on:

- The model.
- The structure of the conversation.
- The length of the context.
- How the application manages the history.

Precisely because this influence varies, it is important to distinguish the chat visible to the user from the context actually supplied to the model.
The user may see a very long conversation on screen, but this does not mean that every message is included in full in every new processing step (Packer et al., 2023).

To stay within context window limits or retain the most relevant content, the application may manage the conversation history in different ways:

- Full inclusion: Previous messages are supplied in their original form.
- Summarization of earlier turns: Older parts are replaced with a summary (Anthropic, 2025).
- Removal: Some messages are excluded from the current context.
- Selective retrieval: Only passages considered relevant are retrieved.
- External memories: Information stored separately from the chat is added when useful (Packer et al., 2023).

These strategies can be combined.
A system might, for example, retain the most recent messages in full, add a summary of earlier turns and retrieve some persistent information about the user or project (cf. Packer et al., 2023).

Consequently, conversational context does not necessarily coincide with the entire history. It is more accurate to define it as the set of information from the dialogue that the application makes available to the model at that moment.

The user cannot normally observe directly which strategy has been used. However, they can derive some clues from the system's behavior.

- If the model accurately recalls a formulation from much earlier, that message might still be present in full or might have been selectively retrieved.
- If it retains only the general meaning but loses examples, exceptions or specific terms, the earlier part might have been summarized.
- If it completely ignores a decision already made, that information might have been removed, not retrieved or made less salient by subsequent elements.

However, these signals do not allow the technical contents of the context to be reconstructed with certainty.

- An imprecise response does not necessarily show that the message has been deleted: the model might have received it without using it correctly (Liu et al., 2024).
- Likewise, a precise response does not show that the original text is still present: the information might come from a summary, an external memory or a new inference.

It is therefore useful to distinguish between actual removal and apparent forgetting.

Actual removal occurs when information is genuinely removed from the context, for example because a new session has been opened without reintroducing it and without any persistent mechanism to retrieve it.
Apparent forgetting, by contrast, occurs when the model seems to have forgotten information that is still available in the context, for example because:

- It is not stated prominently enough.
- It conflicts with subsequent instructions.
- It is buried in a very long conversation.
- It is not recognized as relevant.
- It has been implicitly replaced by a more recent decision.

- It receives less weight than other elements of the context.

When a previous rule is not applied, we therefore cannot immediately conclude that it has been deleted.
It may be necessary to restate it more clearly and declare its priority.

***Example***
Suppose the user says at the beginning:
“When you write for me, always use a formal tone.”

After many turns, the conversation shifts to other aspects: content, examples, structure, corrections, length.

At some point, the user asks:
“Write a brief introduction to this topic.”
The model responds in an informal tone.

At first glance, one might think: “The model has forgotten the rule about formal tone.”
But this is not necessarily the case.
The rule might still be present in the context and might not have been removed. It might simply not have been applied correctly.

For example:
- It is not stated prominently enough: The tone preference appears many turns earlier and has not been mentioned again.
- It conflicts with subsequent instructions: In a recent turn, the user wrote “make it more natural and colloquial”, creating tension with the initial request for formality.
- It is buried in a very long conversation: A large amount of subsequent information may make that rule less salient.
- It is not recognized as relevant: The model may interpret “brief introduction” as a task in which the earlier stylistic preference is not central.
- It has been implicitly replaced by a more recent decision: After several informal texts approved by the user, the system may gravitate toward that style.
- It receives less weight than other elements of the context: A very recent, specific request may have a stronger influence on generation.

If the rule is still available in the context but is not applied correctly, this is apparent forgetting. In this example, this remains a hypothesis: the informal response alone does not allow us to verify whether the rule was actually supplied to the model.

The situation would be different if the rule had genuinely been removed from the context, for example because a new session had been opened without reintroducing it and without any persistent mechanism to retrieve it.
That would be actual removal.

The distinction can be summarized as follows:
- Actual removal: the information is no longer available to the model.
- Apparent forgetting: the information is still available but is not applied or has little influence on the response.

For this reason, if an important rule seems to have been ignored, rather than immediately concluding that it has disappeared, it may be useful to make it explicit again:

“Keep the rule established at the beginning: the tone must remain formal in this introduction too.”

The central point is that presence in the context and actual influence on the response are not the same thing.

To understand what the model seems to have retained, some diagnostic questions can be used:

- What main decisions have we made so far?
- What definitions are you using in this conversation?
- Which previous instructions do you still consider valid?
- Which parts of the work do you consider provisional or outdated?
- Summarize the context you are using to respond.

The answers may reveal omissions, distortions and conflicts.
However, they should not be considered a guaranteed technical representation of the actual context.
The model may produce a plausible reconstruction of what it considers relevant without perfectly describing everything the application has supplied to it (cf. Turpin et al., 2023).

For this reason, the most reliable strategy is not simply to try to discover what remains in the context, but to explicitly reconstruct what must continue to guide the work.

This can be done through formulations such as:

- “Consider the following decisions still valid: …”.
- “The previous definition of X is outdated. From now on, use this new definition: …”.
- “For the next step, use only this information: …”.
- “I am summarizing here the context needed for the new task: …”.
- “The following parts are verified facts; the others remain hypotheses: …”.

This operation can be described as **rewriting the local context.**
It consists of reorganizing the conversation so that the central information becomes visible again, up to date and free of conflicts (cf. Anthropic, 2025).

Rewriting the local context may involve:

- Summarizing conclusions that remain valid.
- Explicitly correcting erroneous premises.
- Revoking outdated instructions.
- Declaring a change of objective.
- Separating facts, hypotheses and decisions.
- Restating fundamental definitions and constraints.
- Removing details that are no longer needed.
- Opening a new conversation when the context is too contaminated.

A new conversation is not always necessary.
It is particularly useful when the previous chat contains too many incompatible instructions, many failed attempts or a framing that continues to steer the work in the wrong direction.

In other cases, an explicit summary establishing which elements to retain and which to abandon may be sufficient (cf. Anthropic, 2025).

The choice depends on the type of continuity needed.

- If the work requires retaining many specific details, it may be preferable to stay in the same conversation and produce an updated summary.
- If, instead, a response needs to be tested without the influence of the previous context, a new session can provide a check that is more independent of the previous conversational context.

Conversational context therefore enables the progressive development of a task, but should not be treated as a neutral, perfect memory.
It may retain useful definitions and decisions, but also errors, outdated objectives and noise. Furthermore, the visible history does not necessarily correspond to what the model receives in full at each turn.

Using it consciously means observing what the system seems to have retained, checking whether important information is still being applied and explicitly reformulating the context when necessary.

Conversational context enables continuity, but also produces inertia.
Managing it well means not only continuing the chat, but updating, correcting and reorganizing what should guide subsequent responses.

# Context compaction and migration

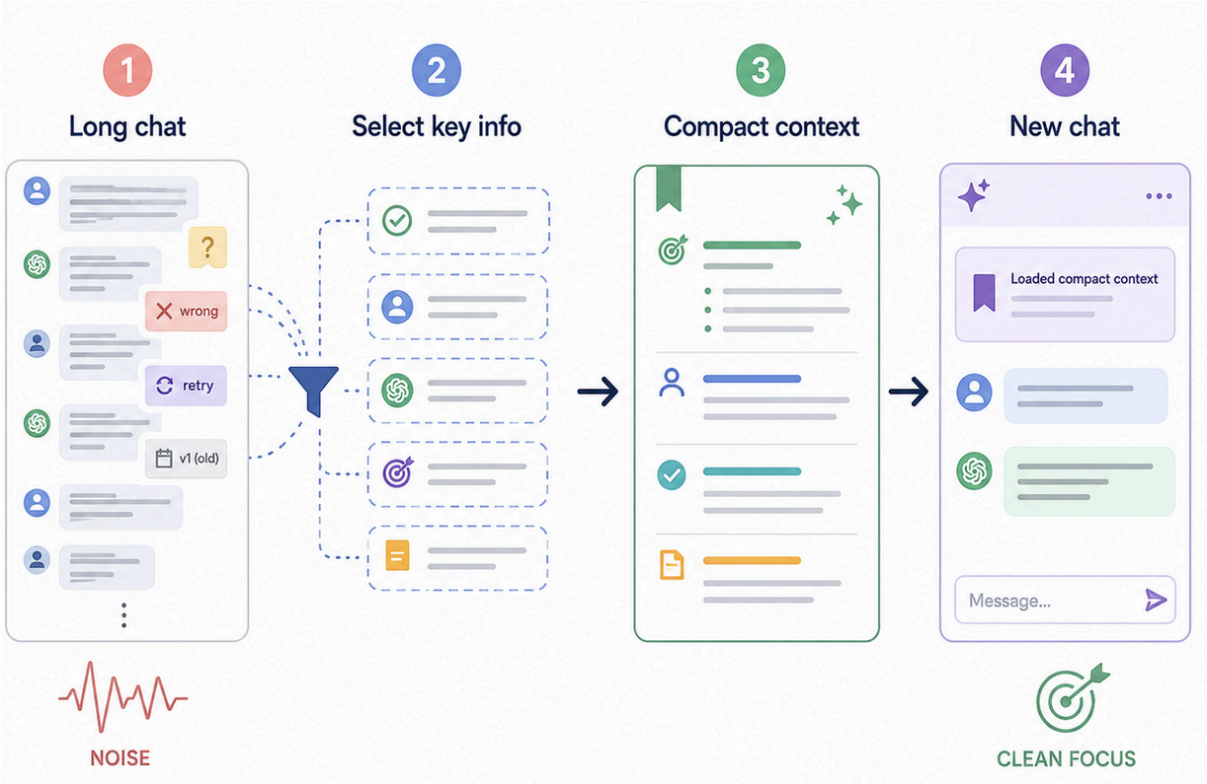
When a conversation becomes very long, the context may accumulate decisions, definitions, corrections, examples, failed attempts and information that is now outdated.
Continuing in the same chat can be useful for maintaining continuity, but it can also increase noise and make the truly important elements less visible.

In these cases, it may be useful to compact the context and, if necessary, migrate it to a new session.

**Compaction** consists of transforming an extensive history into a shorter, organized and updated representation of the work done (Anthropic, 2025).
It does not mean summarizing everything that has been said, but selecting what should continue to influence subsequent responses.

**Migration**, by contrast, consists of transferring this compact representation into a new chat, preserving project continuity without carrying over the entire previous conversation (cf. Anthropic, 2025).

The process can be represented as follows:
Extended conversation → selection of relevant information → structured summary → new conversation




Compaction is particularly useful when the chat contains many repetitions, successive corrections or changes of direction.
In a long conversation, not all information retains the same value. Some decisions remain valid, while others have been superseded; some examples are still useful, while others belonged only to an exploratory phase; some statements have been verified, while others remain mere hypotheses.

A good migration summary should distinguish at least:

- Current objective.
- Decisions that remain valid.
- Definitions and conventions.
- Constraints to observe.
- Relevant materials and sources.
- Results already obtained.
- Open questions.
- Outdated information or information to ignore.

This distinction is important because a summary that mixes final decisions, attempts and hypotheses risks transferring the ambiguities of the previous conversation into the new one (cf. Maynez et al., 2020).

For example, it is not enough to write: “We discussed the project and chose some solutions”.

It is more useful to specify, for example:
“The objective is to design an interface for beginner students. It has been decided to use side navigation and avoid automatic tutorials. The term ‘intuitive’ must be assessed through

completion time, number of errors and ability to complete the first task. The accessibility of the system remains to be verified. Earlier proposals based on exclusively icon-based navigation have been discarded".

This second formulation enables the new dialogue to resume from a clearer state of the work.
Compaction should not be limited to content.
It can also include how the work should be carried out.
It is therefore useful to include the preferences and operating rules that are actually necessary, such as tone, structure, level of detail or verification method.

For example:
"Write in connected prose, with a single main heading. Use lists only when they improve understanding. Always distinguish facts, inferences and assessments. Do not consider a source suggested by the model verified until it has been checked directly."

This makes it possible to migrate not only the project's content but also its instructional context.

The summary can be produced directly by the user or requested from the model.
A useful request might be:
"Prepare a migration summary of this conversation. Include only information that remains valid, distinguishing objectives, decisions, definitions, constraints, results, open questions and outdated content. Do not add new interpretations".

The result must still be checked. The model may omit an important decision, confuse a proposal with a final choice or retain information that is now outdated (cf. Maynez et al., 2020; Anthropic, 2025).
It is therefore useful to verify the summary before using it.

A second step could be:
"Check the summary and indicate which elements derive from explicit decisions, which are inferences and which might be ambiguous."

In this way, compaction also becomes an opportunity to clean up the context.
Once the summary is ready, it can be placed at the beginning of the new conversation together with the new request.
It is preferable to state the text's function explicitly:
"The following block summarizes the context of a previous conversation. Treat it as the basis for the current work. Information marked as outdated must not be used".

The current task can be added after the migrated context:
"Using this context, rewrite the chapter on persistent memory".

This separation between transferred context and new request helps the model distinguish what it needs to know from what it needs to do (cf. Wu et al., 2022).
One possible complete structure is:

```
#Project context
Current objective: ...
Valid decisions: ...
Adopted definitions: ...
Constraints: ...
Available materials: ...
Results obtained: ...
Open questions: ...
Outdated elements: ...

#New task
...
```

Migration does not necessarily have to preserve every detail.
The main criterion is future relevance: information should be transferred if it can change the interpretation of the new task, the choice of sources, the form of the output or the validity of the conclusions (cf. Anthropic, 2025).

- An example discussed many turns earlier can be removed if it is no longer useful.
- A theoretical definition adopted for the entire project, by contrast, must be retained.
- A discarded proposal can be omitted or briefly recorded among the outdated elements when there is a risk of it being proposed again.

It is also useful to distinguish between **complete migration** and **targeted migration**.

- Complete migration transfers the general state of a project and can be useful when opening a new chat to continue the same work.
- Targeted migration transfers only the information needed for a single task. For example, if the aim is to extract only the part concerning editorial tone from a long conversation, there is no need to include all the project's technical decisions.

In general, targeted migration produces cleaner context and reduces the risk of irrelevant information influencing the new output (cf. Shi et al., 2023).

Compaction can also be periodic.
In long projects, it is useful to create intermediate summaries describing the current state of the work.
These summaries can gradually reduce the need to reread the entire history and can be updated when decisions change (cf. Packer et al., 2023; Anthropic, 2025).
The process can take this form:
Conversation → state summary → new developments → summary update

In this way, the context is maintained as a living document rather than a simple accumulation of messages.

There are, however, some risks.

- A summary that is too short may remove important conditions (Anthropic, 2025).
- A summary that is too long may recreate the same overload as the original conversation.

- An overly interpretive formulation may turn hypotheses into decisions (cf. Maynez et al., 2020).
- Completely removing discarded alternatives may lose the reason why a choice was made.

For this reason, good compaction should be:
- Selective.
- Up to date.
- Free of contradictions.
- Explicit about the status of information.
- Proportionate to the new task.
- Checked before migration.

After opening the new conversation, it may be useful to perform a brief check:
“Summarize the decisions and constraints you will use to carry out the new task”.

If the response omits or alters any element, the context can be corrected immediately before proceeding.
This check, too, does not technically reveal everything the system has received or guarantee a reconstruction of its internal processes, but it allows its stated reconstruction to be compared with the decisions and constraints supplied (cf. Turpin et al., 2023).

Migration can also be used to change the phase of the work.
- A chat used for brainstorming can be compacted before starting a new conversation devoted to selection.
- An exploratory conversation can be transformed into a more rigorous context for verification.

In these cases, the previous behavior should not automatically be transferred as well.
For example:
“The previous conversation was exploratory. In the new phase, do not generate further alternatives, but evaluate only the three selected proposals according to the stated criteria.”

Migration therefore makes it possible to preserve useful content without necessarily preserving the same operating mode.

The fundamental principle is that conversational context should not be carried over as a complete copy of the chat, but as a functional representation of the state of the work. Compacting context means preserving what must continue to matter. Migrating it means making that information explicit and usable in a new conversation, without also transferring the noise accumulated in the previous one.

# Supplementary material

## Exercises

**When the conversation continues in the wrong direction**
A group uses a generative system to design a campaign against food waste.
In the first few turns, it is decided to target mainly families with children, and the model proposes numerous ideas based on this audience.
The client subsequently changes the project: the campaign must target restaurant owners exclusively.
The user simply writes:
“Now let’s develop the campaign for restaurants.”

In subsequent responses, however, the model continues to propose educational games, activities to do at home and messages aimed at parents.
Analyze the situation.

- Explain what contextual inertia means.
- Which elements of the previous conversation are probably continuing to influence the model?
- Why does adding a new instruction not necessarily amount to explicitly revoking the previous one?
- How should the change be formulated to make it clear that the previous audience is no longer valid?
- Explain the difference between:
    - Useful reuse of previous information.
    - Propagation of an outdated premise.
- What role can objective inertia play?
- Why might simply continuing the conversation produce further noise?
- Under what conditions would rewriting the local context be sufficient?
- Under what conditions would it instead be preferable to start a new session?
- Explain why conversational context should be considered a resource to update rather than a neutral memory of the work done.

**Has the model really forgotten?**
During a long conversation about designing a game, it had been established that:
“The player must never be able to use weapons.”

After many turns devoted to levels, characters, settings and mechanics, the system proposes:
“In the final level, the player could temporarily obtain a special weapon.”

A student concludes:
“The model has forgotten the rule.”

Critically analyze this conclusion.

- Explain why the observed behavior does not allow us to know with certainty whether the information has actually been removed from the context.
- Distinguish actual removal from apparent forgetting.
- Indicate at least four reasons why a rule that is still present might not sufficiently influence the response.
- Explain why the history visible to the user does not necessarily coincide with the context actually supplied to the model.
- Describe at least four strategies an application can use to manage a very long conversation.
- If the model retains the general meaning of a decision but loses a precise formulation, what possible history management process might have occurred?
- Why does a correct response not show that the original message is still present in full?
- Why does an incorrect response not show that it has been deleted?
- Formulate an intervention that makes the weapons rule explicit and a priority again.
- Explain why asking the model “Which rules do you remember?” can provide useful clues but not a guaranteed technical representation of the context.
- Conclude by explaining the difference between the presence of information in the context and its actual influence on generation.

**Rewriting the local context**
Use a generative system to develop a school podcast project over at least a few turns.
During the conversation, freely introduce:
- Alternative ideas.
- Possible regular segments.
- Episode length.
- Audience.
- Tone.
- Guests.
- Publication frequency.
- Hypotheses still to be decided on.

At some point, change some important decisions and continue the dialogue further.
Then independently analyze the context that has formed.
Your objective is to produce a rewritten local context that allows the project to continue without carrying over outdated or ambiguous elements.

The final submission must distinguish at least:
- Decisions that remain valid.
- Revoked decisions.
- Provisional information.
- Open questions.
- Current constraints.
- Any assumptions introduced by the model.
- Information that is no longer useful for the new task.

After rewriting the context:
- Continue the work using the new version.

- Compare the model's behavior before and after rewriting.
- Identify any problems that have disappeared.
- Identify any important information you removed by mistake.
- Assess whether opening a new chat was really necessary or whether rewriting the local context was sufficient.

Conclude by explaining the criteria you used to decide what to retain, correct, revoke or remove.

**Compacting and migrating a conversation**
Use a conversation that is already sufficiently developed, or build one devoted to designing a short film.

Develop the project until you have accumulated:
- Concepts.
- Characters.
- Narrative choices.
- Production constraints.
- Discarded ideas.
- Corrections.
- Technical decisions.
- Questions still to be resolved.

Once the conversation has become sufficiently complex, you must migrate the project to a new chat.
Do not simply transfer the entire history.
Build a migration summary representing the current state of the project.

Decide independently:
- What must be transferred.
- What can be removed.
- Which discarded alternatives should still be remembered.
- Which decisions should be marked as final.
- Which information should be marked as still provisional.
- Which operating rules should accompany the project into the new conversation.

Then open a new chat, insert the compact context and assign the system a new task consistent with the project.
Compare the behavior of the two conversations and assess:
- Whether the new chat correctly retains important decisions.
- Whether it proposes previously discarded ideas again.
- Whether it distinguishes what has been decided from what remains open.
- Whether compaction has removed useful noise.
- Whether it has instead removed necessary information.
- Whether the new context is too long or too concise.
- What changes you would make to the migration summary.

Conclude by distinguishing compaction from migration and explaining why the main criterion is not to preserve the entire conversation, but to preserve what may change future work.

# Bibliography


Anthropic. (2025, 29 settembre). *Effective context engineering for AI agents*. https://www.anthropic.com/engineering/effective-context-engineering-for-ai-agents

Laban, P., Hayashi, H., Zhou, Y., & Neville, J. (2025). *LLMs get lost in multi-turn conversation* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2505.06120

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., & Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12, 157–173. https://doi.org/10.1162/tacl_a_00638

Maynez, J., Narayan, S., Bohnet, B., & McDonald, R. (2020). On faithfulness and factuality in abstractive summarization. In *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

Packer, C., Wooders, S., Lin, K., Fang, V., Patil, S. G., Stoica, I., & Gonzalez, J. E. (2023). *MemGPT: Towards LLMs as operating systems* [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2310.08560

Shi, F., Chen, X., Misra, K., Scales, N., Dohan, D., Chi, E. H., Schärli, N., & Zhou, D. (2023). Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning* (Vol. 202, pp. 31210–31227). PMLR. https://proceedings.mlr.press/v202/shi23a.html

Turpin, M., Michael, J., Perez, E., & Bowman, S. R. (2023). Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems* (Vol. 36). https://proceedings.neurips.cc/paper_files/paper/2023/hash/ed3fea9033a80fea1376299fa7863f4a-Abstract-Conference.html

Wu, T., Terry, M., & Cai, C. J. (2022). AI Chains: Transparent and controllable human-AI interaction by chaining large language model prompts. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (Articolo 385, pp. 1–22). Association for Computing Machinery. https://doi.org/10.1145/3491102.3517582


# Further reading

- Mike Taylor, Context Engineering with DSPy, O'Reilly Media, 2026 — Probably the closest overall match to this handout. It treats context as an engineered resource and examines long-context failure modes, context distraction, context drift, context underspecification, context editing, summarization, memory offloading, retrieval, guardrails, and control flow. It is particularly relevant to the handout's treatment of long conversations as contexts that must sometimes be reorganized rather than merely extended. oreilly.com

- Boni García, Context Engineering: Build Consistent, Accurate, Predictable AI Systems, Manning Publications, 2026 — A very strong complement for understanding context selection, organization, state, memory, retrieval, and agent orchestration. It is especially useful for extending the handout's distinction between information that remains in the current conversation and information that is stored or retrieved through broader system mechanisms. The book is particularly aligned with the idea that useful context must be actively maintained rather than treated as an unlimited transcript. Manning Publications
- Chip Huyen, AI Engineering: Building Applications with Foundation Models, O'Reilly Media, 2024 — Provides the broader systems perspective needed to understand why conversational behavior depends on more than the model itself. It covers foundation-model applications, conversational systems, evaluation, retrieval, agents, and application architecture. It is especially relevant to the chapter's distinction between the conversation visible to the user and the information actually supplied to the model by the application. oreilly.com
- Valentina Alto, Building LLM Powered Applications: Create Intelligent Apps and Agents with Large Language Models, Packt Publishing, 2024 — Particularly useful for the practical side of conversational applications and memory. Its conversational-application material explicitly deals with chatbot schemas, memory components, external knowledge, and tools. It therefore provides a concrete technical continuation of the handout's discussion of conversation history, external memory, and state management. packtpub.com
- Micheal Lanham, AI Agents in Action, 2nd ed., Manning Publications, 2026 — A useful advanced continuation for understanding stateful agentic workflows, where a system must maintain information across multiple operations, use tools, consult knowledge, and coordinate actions over time. It is less specifically about ordinary chat history than the first two books, but highly relevant to the broader question of how context, memory and state are managed when conversational systems become persistent agents.

# Editorial notes

In producing these materials, I have used generative artificial intelligence tools to support the writing process, particularly to improve the form of the text, reorganize content, refine wording, and speed up certain editorial tasks.

As I work independently on the production of these materials, I try to automate any activities that can reasonably be automated, so that I can devote more time to research, design, and content development.

However, artificial intelligence does not determine the content of this work: the choice of topics, structure, ideas, interpretations, examples, and teaching approach are developed by me. AI is therefore used primarily as a tool to support production and formal revision, while authorship and responsibility for the design and content remain mine.

Unless otherwise indicated, this material is distributed under the Creative Commons Attribution–NonCommercial–ShareAlike 4.0 International license (CC BY-NC-SA 4.0).

You are therefore free to share, redistribute, adapt, and create derivative works based on this material, provided that appropriate credit is given, the material is not used for commercial purposes, and any modified or derivative versions are distributed under the same license.

