Context Engineering, Prompt Engineering, and Semantic Fidelity

Why Better Context Does Not Always Produce Better Understanding

Representational Systems Brief #5

A. Jacobs | Semantic Fidelity Project

The Basic Pattern

Large language models reason from the information placed in front of them rather than from direct contact with the world. That information may include a user's prompt, retrieved documents, earlier interactions, or outputs from external tools. Context engineering shapes how these materials are selected and assembled before reasoning begins.

These representations make modern AI remarkably capable. Rather than relying solely on pretrained parameters, models increasingly reason over dynamically assembled contexts that combine retrieval, memory, planning, and external tools.

Over time, however, a familiar pattern can emerge. The context becomes larger, retrieval becomes more sophisticated, and more information becomes available to the model, yet the quality of understanding does not necessarily improve. Context remains complete while meaning gradually becomes more difficult to preserve.

Artificial intelligence describes these systems using terms such as context engineering, prompt engineering, retrieval augmentation, context windows, memory management, and tool orchestration. Each addresses different aspects of supplying information during model reasoning. Within the Semantic Fidelity framework, these are technical examples of a broader representational challenge.

More Context Is Not More Understanding

Context engineering expands what a model can reason over by bringing documents, retrieved passages, conversation history, memory, and tool outputs into the same working environment. This makes richer reasoning possible, but it also increases representational complexity. When optimization outpaces constraint, meaning becomes optional.

As the context grows, information must be selected, ranked, summarized, reordered, and integrated. Each transformation creates another opportunity for important relationships, priorities, or constraints to weaken.

The central question is therefore not whether the context window is large enough, but whether the representations inside it still preserve the meaning required for correct reasoning.

Understanding the Problem

Context Engineering: The design and management of the information presented to AI systems during reasoning.

Prompt Engineering: The construction of prompts that guide model behavior and responses.

Context Windows: The amount of information a model can process during a single reasoning cycle.

Retrieval Augmentation: The process of adding external information into the reasoning context.

Although these approaches differ technically, they share a common objective. They improve the quality of representations available to the model. They do not automatically preserve the meaning carried by those representations.

How Context Drift Emerges

Stage 1 — Representation

Information is collected from prompts, memory, retrieval systems, tools, and external documents.

Stage 2 — Selection

Relevant information is chosen, ranked, summarized, and assembled into a reasoning context.

Stage 3 — Integration

The model combines multiple representations into a coherent explanation or decision.

Stage 4 — Semantic Drift

The reasoning may remain coherent even as the relationships within the context begin to weaken. More information is present, but it becomes harder to preserve how the pieces fit together. Nothing may be obviously missing, yet the context can still become increasingly difficult to interpret correctly.

Semantic Fidelity and Reality Drift

Context engineering focuses on improving the information available to the model. Semantic fidelity asks whether that information still preserves the assumptions, distinctions, relationships, and constraints required for correct interpretation. A model may receive every relevant document and still reorganize them into a different conceptual structure. Retrieval can succeed, the context can appear complete, and the reasoning can remain fluent even as meaning changes through integration rather than omission.

Reality Drift places this problem within a broader pattern. As AI systems increasingly reason through assembled representations rather than direct observation, maintaining a coherent context can become easier than maintaining correspondence with the reality that context was meant to describe.

The Context Stack

Modern AI systems increasingly reason through layered representations. Reality is captured as observations, turned into documents, selected through retrieval, assembled into context, and transformed into reasoning and responses. Those responses may then become part of future context, allowing the cycle to continue.

Each layer makes reasoning more flexible, but it also creates another opportunity for meaning to change during transformation. A system may possess all of the relevant information while gradually losing the relationships that make that information meaningful. The further reasoning moves from direct observation, the more important semantic fidelity becomes

Examples Across AI Systems

Enterprise AI: Multiple company documents are assembled into a context window, yet differences between departments, policies, and historical assumptions become compressed into a single generalized answer.

Research Assistants: Relevant papers are successfully retrieved while conflicting methodologies and definitions become less explicit during synthesis.

Coding Agents: Project documentation, code history, and previous conversations are all available, yet important design assumptions disappear as context is repeatedly summarized and reused.

Legal AI: Cases, statutes, and regulations are retrieved successfully while jurisdictional distinctions gradually weaken during explanation.

Personal AI Assistants: Long conversation histories remain available while repeated summarization gradually changes the user's original intentions.

Semantic Fidelity as a General Principle

Context engineering represents an important evolution in artificial intelligence because it improves how information is assembled before reasoning begins. It determines which documents, memories, instructions, and external sources become available to the model, while prompt engineering helps shape how that information is used. Semantic fidelity addresses a different layer of the problem by asking whether the meaning carried by those representations survives selection, compression, ordering, integration, and reasoning.

As AI systems depend more heavily on dynamic context rather than static training alone, providing the right information is no longer enough. A context may appear complete while the relationships, distinctions, and assumptions within it gradually change. The deeper challenge begins after the context has been assembled. Maintaining semantic fidelity requires preserving meaning as information moves through increasingly complex reasoning systems, making this as important as retrieval itself.

Related Concepts

Context Drift: The gradual weakening of meaning as information is selected, reordered, summarized, and integrated into a model's working context.

Interpretation Failure: A condition in which the necessary information is present but is organized or understood in a way that changes its intended meaning.

Recursive Compression: The repeated summarization of already compressed context, increasing the risk that distinctions disappear while coherence remains.

Representation Drift: The gradual change in what prompts, retrieved passages, memories, or tool outputs mean as they move through the reasoning process.

Reality Drift: The broader condition in which a system remains coherent and operational while its assembled representations lose alignment with the reality they were meant to describe.

Frequently Asked Questions

Is context engineering the same as prompt engineering?

No. Prompt engineering focuses on how instructions are written, while context engineering manages the broader set of information available during reasoning, including memory, retrieval, documents, and tool outputs.

Does a larger context window improve understanding?

Not automatically. A larger window can provide more information, but it can also make relationships, priorities, and constraints harder to preserve.

Can retrieval succeed while reasoning still fails?

Yes. The correct documents may be retrieved, yet the model can still compress distinctions, misread relationships, or reorganize the information into a different conceptual structure.

How does semantic fidelity apply to context engineering?

Semantic fidelity asks whether meaning survives the selection, compression, ordering, and integration of information into the reasoning context.

Related Search Terms

- context engineering
- prompt engineering
- context windows
- retrieval augmentation
- retrieval-augmented generation
- GraphRAG
- memory management
- tool orchestration
- context compression
- semantic search
- semantic fidelity
- interpretation failure
- representation drift
- reality drift
- grounding
- long-context models