# LLM PITFALLS

SEVERITY LOW ▼ ● ▲ HIGH

| CATEGORY | PITFALL | | MITIGATION |
|---|---|---|---|
| Design | Using generative AI as a first resort | ▲ | Consider more explainable, scrutable tech |
| | No baseline or state-of-the-art performance | ● | Measure SOTA performance |
| | No success criterion | ● | Ask user/customer what success means |
| Instruction following | Refusal to answer on unreasonable grounds | ▼ | Add context, remove ambiguity |
| | Misinterpretation, failure to clarify | ▲ | Instruct to clarify, add context, be explicit |
| | Failure to reject false or inconsistent query | ● | Avoid incorrect or inconsistent prompts |
| | Arbitrary execution of instructions in data | ▲ | Explicitly separate instructions and data |
| | Difficulty manipulating sub-token elements | ● | Pre-process sub-token elements |
| | Incorrectly formatted output | ● | Provide format schema or template |
| Ambiguity & coherence | Ambiguity in the response | ▲ | Prompt engineering, pre-check responses |
| | Incoherent rambling or glitching | ▼ | Reword prompt and retry |
| | Inconsistent reasoning across conversation | ● | Shorten conversations, provide recaps |
| | Irrelevance | ▼ | Prompt engineering, pre-check responses |
| Response quality | Simplistic (correct but poor) responses | ● | Role allocation, few-shot prompts |
| | Incompleteness | ● | Role allocation, few-shot prompts |
| | Faulty reasoning | ▲ | Chain-of-thought, tools, few-shot prompts |
| | Faulty premises | ▲ | Explicit chain-of-thought, break steps down |
| | Prompt sensitivity | ● | Add more context and more instruction |
| | Overthinking | ● | Provide more context about problem domain |
| | Distraction | ▼ | Use neutral wording |
| | Verbose | ▼ | Specify minimal requirements |
| Factual errors | Incorrect factual recall | ▲ | RAG, knowledge graphs, pre-check response |
| | Fabricated or non-useful references | ▲ | RAG, knowledge graphs |
| | Coherent fiction | ▲ | RAG, knowledge graphs, pre-check response |
| | Out-of-date responses | ● | RAG, tools |
| | Overconfidence in certainty | ● | Disallow expressions of uncertainty |
| Bias & ethics | Explicit bias | ▲ | Improve alignment and system prompt |
| | Implicit bias | ▲ | Improve alignment, avoid leading signals |
| | Toxic or offensive responses | ▲ | Pre-check response |
| | Copyright infringement | ▲ | Knowledge graphs |
| | Cultural or ideological bias | ● | Fine-tuning, alignment |
| Application | No transparency about role of AI | ▲ | Be clear |
| | Not collecting signals from users | ▼ | Collect signals |
| | Not providing references | ● | Add references |
| Implementation | No rigorous evaluation protocol | ▲ | Create testing and reporting pipeline |
| | Low AI literacy among users | ● | Education and culture |
| | Overreliance on model responses | ▲ | Education, avoiding risky applications |
| | Overreliance on human-in-the-loop | ● | Consultation, training, improved interface |
| | No user training | ● | Training |
| | No consideration of security or ethics | ▲ | Proper governance and oversight |



mtha@equinor.com

A highly opinionated and non-exhaustive list of pitfalls aimed at developers of applications containing large language models,. There are more than 30 concerns here, but some of them overlap, and probably still others are missing entirely. This list is my own opinion, which is not necessarily shared by others at Equinor or anywhere else. It is a work in progress, you input will be welcomed and credited, please get in touch!

Abbreviations: RAG means 'retrieval augmented generation'.