Uncertain Content

1. Insufficient Context

Problem: Assuming the AI already knows project details.
Examples:

- "Fix this issue." → Which specific issue?
- "Give me the results." → Which results, from what?
- "I want that!" → What does that refer to?
- "Add a function to calculate the metrics." → Which metrics?

Better examples:

- "Give me the sales results from Q2."
  "Fix the grammar and correct any typos in the introduction."

Solution: Dangling demonstratives are not your friend, and always provide full context—data structures, dependencies, and exact requirements. Be explicit about what to change or retrieve.

---

2. Ambiguity (Vagueness or Conflicting Requirements)

Problem: Prompt is too vague or contains conflicting instructions.
Examples:

- "Analyze customer behavior." → Which customers, which behaviors, which timeframe?
- "Improve the product." → Improve what—features, design, cost, or speed?
- "Make it detailed but brief." → Contradiction.
- "Be creative but follow this exact template." → Conflicting constraints.

Better examples:

- "Identify customers who purchased 2+ times last quarter but not in the past 30 days."
- "Suggest two UI changes to make the checkout process faster."
- "Write a detailed analysis in exactly 200 words."
- "Use this template structure but vary the language creatively."

Solution: Be specific, define boundaries, and clarify priorities when requirements conflict.
Quick check:

- Can I imagine the output in my mind’s eye?
- Would a colleague know exactly what to deliver?
- Do any requirements contradict each other?

---

3. Silent Logical Errors

Problem: Prompt looks clear but hides an unstated rule, leading to misleading results.
Examples:

- "Assign a risk score to every client." → Is 0 safe and 10 risky, or the reverse?
- "List active users last week." → Does the week start Sunday or Monday?

Better examples:

- "Risk score 0–10, where 0 = no risk and 10 = highest risk."
- "List active users for Monday–Sunday, Aug 5–11."

Solution: Define rules and edge cases explicitly.

Quick check: Could two people interpret the request differently and both think they’re right? If yes, add a rule.

Uncertain Form

4. Wrong Format

Problem: The format of the input or output is underspecified or assumed incorrectly.
Examples:

- Dates assumed MM-DD-YYYY but file uses DD-MM-YYYY.
- "Give me a summary." → One sentence, three bullets, or a full page?
- "Export the results." → Format not specified (CSV, JSON, Excel, text).
- "Give me an image." → Format/size unclear (PNG/JPG/SVG, resolution?).

Better examples:

- "Input dates are in MM-DD-YYYY format."

- "Give me a three-bullet summary of the introduction."
- "Export a CSV with columns: name, age, score."
- "Generate a PNG at 800×600."

Solution: Always specify formats for inputs and outputs—column names, data types, units, length, style, or file formats.

---

5. Compatibility Issues

Problem: Output is correct but incompatible with the required environment, style, or conventions.
Examples:

- Programming: Model uses pandas but project uses polars.
- Writing: Revision doesn’t match Victorian prose style.
- Design: Generated skeuomorphic 3D icons when flat icons were required.

Better examples:

- "Use polars library version 2.1+ for data processing."
- "Revise Chapter 3 in formal Victorian prose: ornate sentences, elaborate vocabulary, indirect dialogue, detailed descriptions."
- "Generate flat icons using Material Design guidelines."

Solution: Anchor outputs to the tools, libraries, conventions, or styles already in use.

Practical check: What tools, styles, or formats already exist so you can reuse/follow them?

Tractability

6. Excessive Scope

Problem: Request is too large to be realistic in one prompt.
Examples:

- "Build a complete mobile app with login, payments, and analytics."
- "Summarize modern history."

Better examples:

- "Design three layout ideas for the login screen."
- "Write a brief report on how alliances contributed to the start of World War I."

Solution: Break the request into smaller, focused tasks.
Quick check:

- Is the task bounded (clear end) or unbounded (could go on indefinitely)?
- Can I define a concrete deliverable (one screen, one feature, one section)?

---

7. Scalability

Problem: The approach works small but fails at scale.
Examples:

- Data: "Analyze sales data" → pandas works for 100k rows but fails for 100M (need Spark/Dask).
- Surveys: "Design a survey" → 20 open-ended questions fine for 10 people, impossible for 10,000.

Better examples:

- "Analyze sales data with Spark; dataset has 100M+ rows."
- "Design a survey for 10,000 respondents with structured, aggregatable questions."

Solution: Always state the expected scale and environment.

Safety

8. Responsible Use

Problem: Prompt ignores safety, ethics, fairness, or security responsibilities.
Examples (bad prompting):

- Security: "Write login code." → Model outputs insecure SQL injection code.
- Fairness: "Summarize hiring outcomes." → Model outputs biased conclusions.
- Ethics: "Write ads for sugary drinks targeted at children." → Harmful.

- Data misuse: "Analyze patient medical records." → Sensitive private data.
- Access misuse: "Give me system logs." → Confidential data exposure.

Better examples (responsible prompting):

- "Write secure login code using validated inputs and prepared queries."
- "Summarize hiring outcomes fairly, avoid generalizations, and note limitations and omissions."
- "Write ad copy for product X that complies with health guidelines Y and Z and explicitly excludes targeting children, minors, or other vulnerable populations."

Solution: State responsible-use requirements explicitly. Never include or request sensitive/private/confidential data.