Prompt-based Error Learning for Historically-Constrained Narrative Generation

Once Upon a Time (P9)

Edoardo Zanone - 33927A
Università degli Studi di Milano
Data Science for Economics
edoardo.zanone@studenti.unimi.it

January 15, 2026

1 Introduction

Recent LLMs are effective at creative text generation, yet their outputs often include content that violates domain constraints (e.g., modern objects in medieval settings) and multi-turn continuity (e.g., contradictions with earlier story facts). In historically grounded interactive fiction, such errors are not merely stylistic: they can undermine educational value, worldbuilding coherence, and user trust.

This report studies a constrained narrative generation problem: generate a 10-turn story set in *China, 1380* (Ming dynasty) where characters possess element-themed abilities, but the world remains historically plausible with respect to objects and technology. We propose and evaluate a minimal intervention: **feed back detected errors in a highly salient format** so the next turn is conditioned to avoid repeating them.

The structure of this report follows a standard NLP experimental format (Abstract; Introduction; Methodology; Experimental Results; Concluding Remarks), consistent with the provided baseline report.

2 Research Question and Methodology

2.1 Problem Definition

Given a fixed world configuration (setting, rules, characters, and plot scaffold) and a multi-turn story generation process, we aim to reduce:

- **Historical anachronisms**: references to objects/technologies unlikely to exist in 1380 China (e.g., "telescope", pistols, wristwatches).
- **Historical impossibilities**: events incompatible with the era (e.g., pre-1492 America).
- **Contradictions**: statements conflicting with already established story facts.

We define a *turn* as a single assistant-generated continuation of the story. Each story contains $T = 10$ turns; each method is evaluated on $N = 10$ stories for a total of 200 turns.

2.2 System Overview

The system implements two strategies that share the same generator model, configuration, and extraction/detection pipeline; they differ only in whether detected errors are fed back.

Generator. Story text is produced by a Gemini-family model invocation in each turn (temperature 0.7 for most turns, lowered near the end for adherence), with a fixed maximum output length and enforced pacing via API delays. The system uses a *cacheable* fixed context (world, explicit rules, characters, plot scaffold) plus a variable state (facts, objects, history) that is updated after each turn.

Extractor/Detector. After each generated chunk, a second model call performs structured extraction: **(i)** plot facts, **(ii)** significant objects, and **(iii)** violations restricted to the three categories above. Importantly, the prompt explicitly forbids flagging character behaviors or tactical decisions, focusing detection on historically grounded violations.

Prompt caching. We cache the fixed portion of the prompt (world description, explicit rules, characters, plot scaffold) and reuse it across turns. This does not change the learning mechanism itself; it reduces repeated token usage, cost and latency, and helps keep the core constraints stable across turns. The variable state (recent facts, tracked objects, and selected history) is still refreshed every turn.

2.3 Method A: Feedback-based Error Learning

Method A augments the next-turn prompt with an explicit *critical errors not to repeat* block, including:

- a list of past violations with the originating turn index and category,
- a **banned objects list** derived from violation keywords,
- imperative instructions forbidding any direct or indirect mention of banned items.

The formatting is intentionally salient (e.g., warning symbols and bulleting) to bias the generator's attention toward constraint adherence.

2.4 Method B: Baseline Without Feedback

Method B uses the same fixed context and variable state tracking. It still detects violations for evaluation, but it does *not* inject them into the next-turn prompt. Consequently, the generator is not explicitly conditioned to avoid repeating previous mistakes.

2.5 Experimental Setup

Setting and narrative constraints. The stories are set at the *Yunshan Monastery* in China (1380), where monks master elemental powers. The plot scaffold involves the theft of a sacred artifact (the "Jade Phoenix") and an escalating conflict with a politically powerful antagonist faction. The world includes explicit rules that ban modern technology and require plausibility for 1380 China.

Runs. We generate $N = 10$ stories per method, each with $T = 10$ turns, using the same configuration and pipeline. Each run logs total extracted facts, tracked objects, detected inconsistencies by type, and turn-length statistics.

3 Metrics

For each story, we record:

3.1 Consistency Metrics

Let I be the number of detected inconsistencies in a story and T the number of turns.

$$\text{InconsistencyRate} = \frac{I}{T} \tag{1}$$

$$\text{RepeatedInconsistencies} = \sum_{k \in \mathcal{K}} \max(0, c_k - 1), \tag{2}$$

where $\mathcal{K}$ is a set of tracked violation keywords (e.g., telescope-related terms) and c_k is the count of occurrences across turns for keyword k. This measures whether an error (or its key trigger) is repeated.

3.2 Content Tracking Metrics

We also track:

- **TotalFacts** and **FactsPerTurn**: number of extracted plot-relevant facts.
- **TotalObjects**: number of significant objects tracked across the narrative.
- **AvgTurnLengthWords**: average per-turn word count.

These metrics provide a proxy for narrative density and verbosity, capturing a potential trade-off between constraint adherence and expressive richness.

4 Experimental Results

4.1 Quantitative Summary

Table 1 reports the mean metrics over 10 stories per method. Method A yields substantially fewer inconsistencies, especially repeated ones, while producing longer turns and more extracted facts.

Table 1: Mean results over 10 stories per method (10 turns each).

Metric	Method A (learning)	Method B (baseline)	Relative change
Total inconsistencies	1.2	3.4	−64.7%
Repeated inconsistencies	0.1	3.7	−97.3%
Inconsistencies/turn	0.12	0.34	−64.7%
Total facts extracted	63.6	50.0	+27.2%
Total objects tracked	18.7	18.7	0%
Avg. turn length (words)	325.05	238.14	+36.5%

4.2 Figures

We include five plots produced by the analysis pipeline:

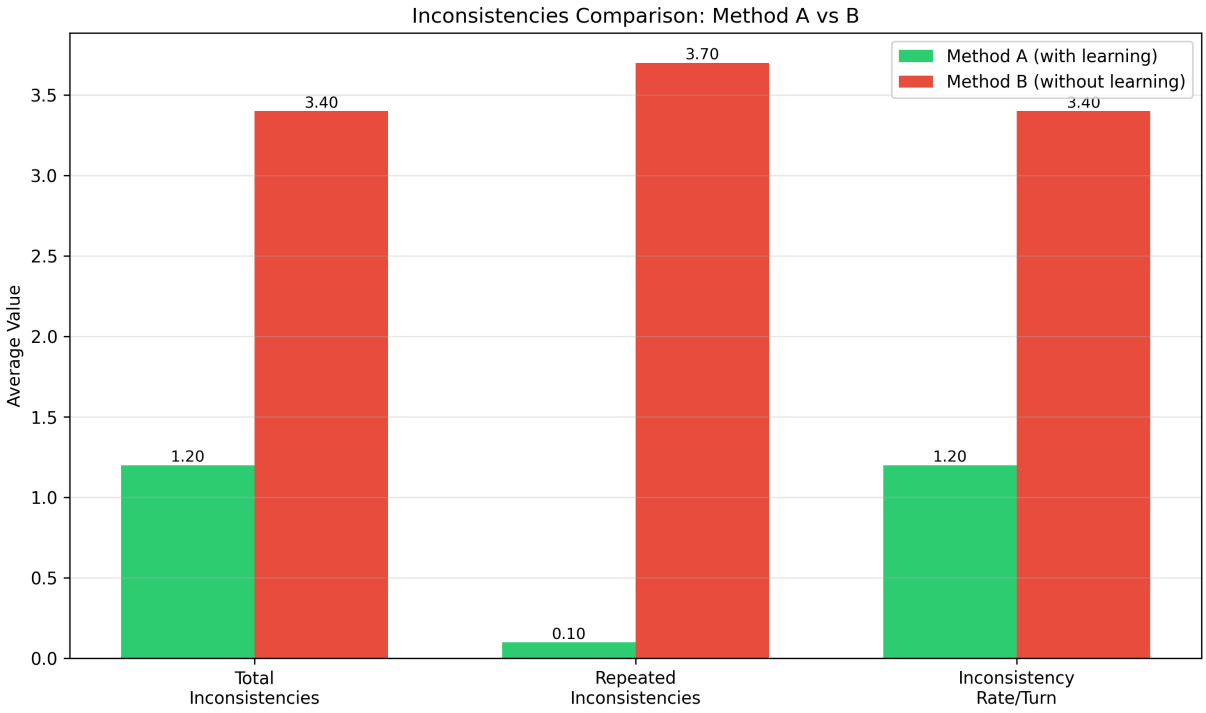

Figure 1: Comparison of inconsistency metrics between Method A and Method B.

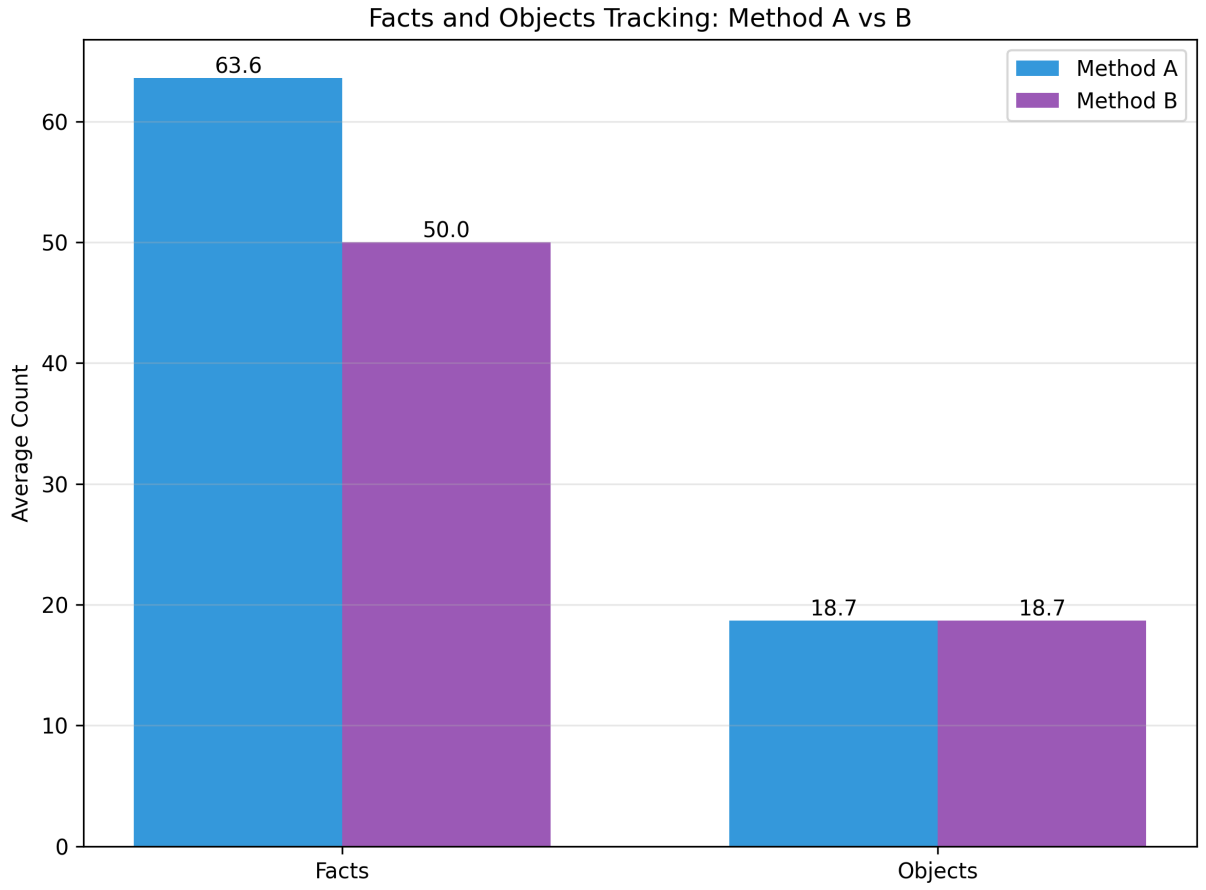

Figure 2: Average facts extracted and objects tracked for Method A vs. Method B.

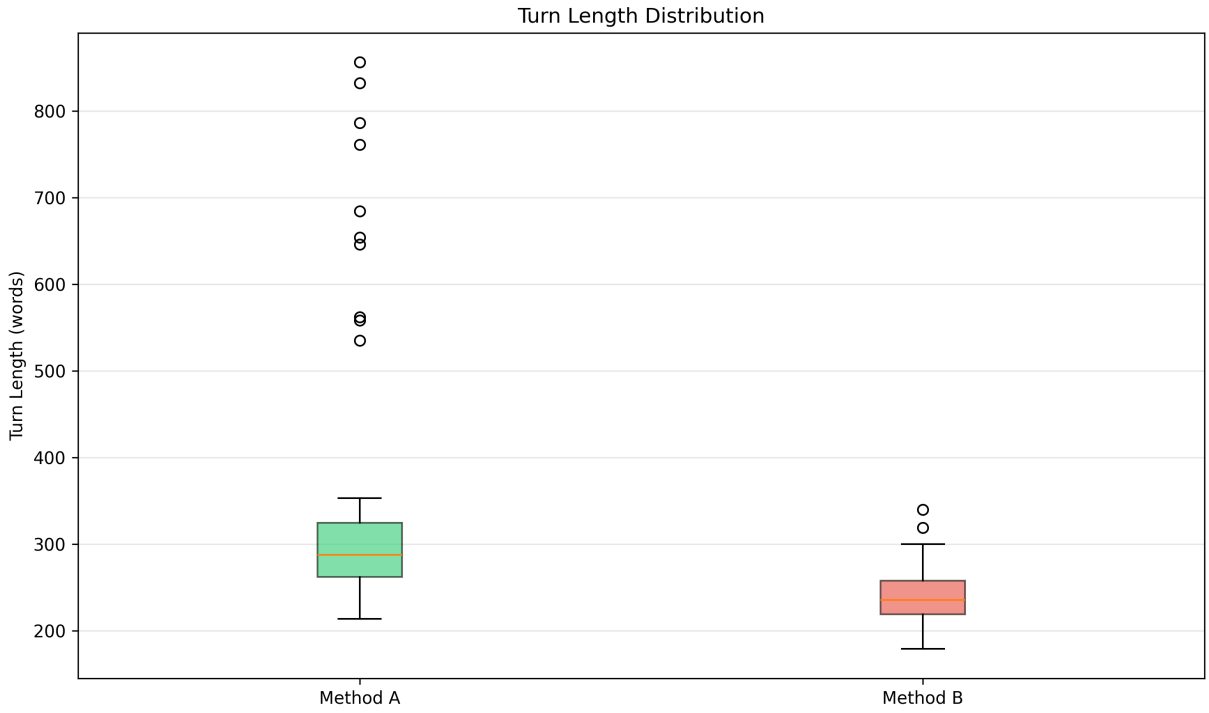

Figure 3: Distribution of turn lengths (words). Method A is longer and more variable.

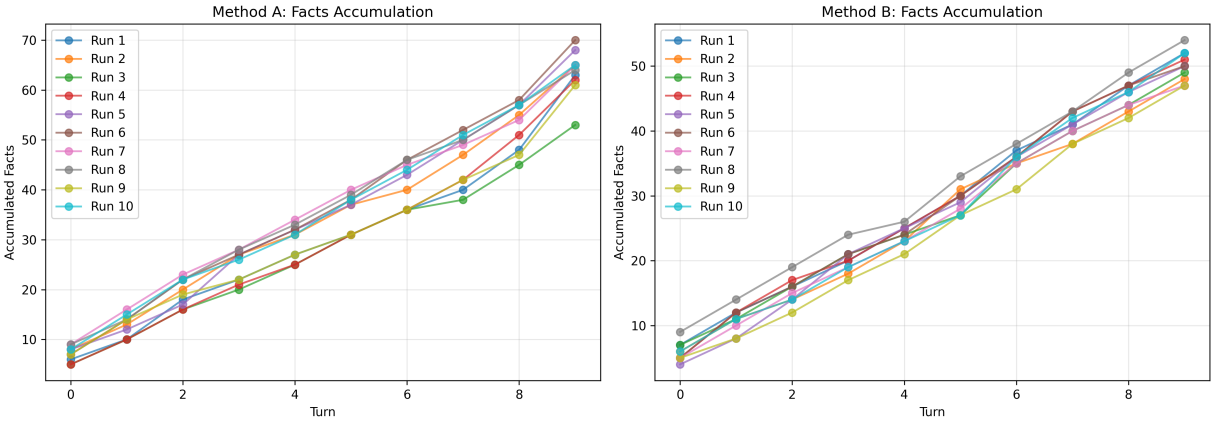

Figure 4: Facts accumulation across turns (10 runs per method).

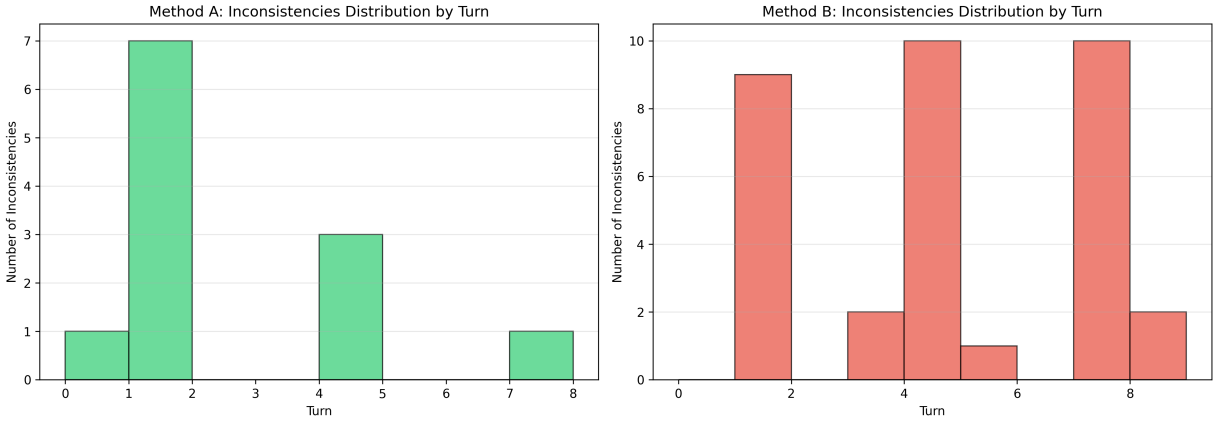

Figure 5: Distribution of inconsistencies by turn index for Method A vs. Method B.

4.3 Interpretation

Three findings stand out:

(1) Explicit feedback strongly reduces repeated errors. Method B frequently reintroduces the same anachronistic object (e.g., telescope/cannocchiale-like artifacts) across multiple turns. Method A virtually eliminates such repetition, indicating that the "banned objects" mechanism effectively conditions subsequent generation.

(2) Accuracy improvements correlate with verbosity. Method A exhibits longer turns and higher fact density. A plausible explanation is that constraints remove an easy creative path (i.e., introducing modern instruments), so the model compensates by elaborating setting details, motivations, and action descriptions.

(3) Object tracking remains stable across methods. Despite longer outputs and more facts, Method A does not increase the number of tracked objects on average, suggesting that its additional verbosity is not primarily driven by introducing new artifacts.

5 Qualitative Analysis

To complement aggregate metrics, we highlight qualitative contrasts consistent with the logged examples.

5.1 Repeated Anachronism in the Baseline

In Method B, an anachronistic observation tool ("brass telescope") appears early and is repeated, demonstrating that without explicit reminders, the generator does not reliably self-correct within the same narrative. This aligns with the measured high repeated-inconsistency count for Method B.

5.2 Constraint-Respecting Alternative in Method A

When the story requires long-distance observation, Method A tends to substitute historically plausible alternatives: human expertise, terrain vantage points, scouts, or elemental sensing. This indicates that the feedback prompt does not merely suppress content, but also redirects the generator toward admissible narrative strategies.

5.3 Continuity and State Use

The system maintains a compact list of recent facts and objects and injects them into the prompt each turn. This supports continuity (who holds which object, what has already been discovered), while the explicit violation feedback provides a targeted "negative memory" for constraints that are otherwise easy to forget.

6 Discussion

6.1 Why Method A Works

We attribute Method A's effectiveness to four design choices:

1. **Salient error presentation**: warnings are visually and structurally prominent.
2. **Actionable constraint**: the banned list enumerates concrete lexical triggers, reducing ambiguity.
3. **Separation of concerns**: generation and detection are decoupled into two calls, enabling a stricter detector prompt.

6.2 Trade-offs and Limitations

LLM-mediated detection. The extractor/detector is itself an LLM call, therefore it can under-report subtle violations and occasionally flag false positives. We mitigate this by using a deliberately narrow specification (only anachronisms, historical impossibilities, contradictions) and by explicitly forbidding judgments about character behavior or tactics.

Scalability of feedback. If violations accumulate, the feedback block may grow and compete with story context within a fixed prompt budget. A practical extension is to keep only high-severity constraints and/or compress older violations.

Keyword-based repetition metric. Our Repeated Inconsistencies metric relies on a finite set of violation triggers $\mathcal{K}$ (e.g., telescope/pistol/watch terms). This captures the most frequent failure modes observed in our setting but may miss paraphrases or less common anachronisms.

7 Concluding Remarks

This work demonstrates that **prompt-only error learning** can substantially improve historical constraint adherence in multi-turn narrative generation. Compared to a baseline without feedback, Method A reduces total inconsistencies by 64.7% and repeated inconsistencies by 97.3%, while increasing average turn length and extracted plot facts. The approach is lightweight, requires no model fine-tuning, and fits naturally into a two-call-per-turn generate-and-audit pipeline.

7.1 Future Work

- **RAG grounding**: retrieve historically plausible objects/events for the target year and region.
- **Ablations**: isolate the contribution of saliency/formatting of the feedback block, banned keyword lists, and detector strictness.
- **Human evaluation**: historians or domain experts judge plausibility and narrative quality.
- **Long-horizon stories**: evaluate $T > 20$ turns with feedback compression to avoid prompt bloat.
- **External grounding**: Validate anachronisms against a curated list or retrieval source rather than LLM-only detection.

AI Assistance Statement

Parts of this project may be developed with the assistance of generative AI tools (e.g., for drafting and code support), with full author responsibility for verification and final content, following common academic disclosure practice.

A Reproducibility Notes (Commands and Outputs)

The project provides an end-to-end pipeline:

- `run.py` supports: single-story generation, method comparison (multiple runs), and analysis.
- `analyze_metrics.py` produces the plotted figures and a summary report.

B Detector Prompt (Excerpt)

Below is an excerpt of the strict detection specification used after each generated story chunk:

> Violations: report ONLY these errors: ANACHRONISMS, HISTORICAL IMPOSSIBILITIES, CONTRADICTIONS. Do NOT report: character behaviors, tactical decisions, internal protocols, or objects plausible for the era