Backstory–Narrative Consistency Verification System

Track A: Systems Reasoning with NLP & Generative AI

Team Name: The Dream Team

Competition: Kharagpur Data Science Hackathon 2026 – Track A

Submission Type: Round 1 Final Report

Submitted By:

Shivam Shreshtha (Team Leader)

Kritika

Navnita

Sneha

Abstract

Understanding whether a character’s hypothetical backstory is consistent with a long-form narrative requires more than simple fact matching. It demands reasoning across time, events, motivations, and causal structure embedded throughout the text. In this work, we present a scalable and explainable system for verifying backstory–narrative consistency in full-length novels (100k+ words). The system treats the novel as ground truth and the backstory as a hypothesis, decomposes the backstory into atomic claims, retrieves supporting narrative evidence from the novel, and produces a deterministic binary verdict. Our approach emphasizes clarity, long-context handling, and evidence-based reasoning, aligning with Track A’s focus on systems reasoning.

Table of Contents

1. Introduction

Narratives in long-form fiction encode character development implicitly across actions, dialogue, and evolving motivations. When an external backstory is proposed for a character, verifying its consistency with the full narrative becomes a non-trivial reasoning task. This problem is especially challenging for long texts where relevant evidence may be distributed across thousands of pages.

The goal of this system is to automatically determine whether a given hypothetical backstory aligns with the narrative constraints of a complete novel. Our solution prioritizes interpretability and robustness over black-box prediction, making each decision traceable to concrete narrative evidence.

2. Problem Definition

Inputs

- A full-length novel (100k+ words)
- A hypothetical backstory describing motivations, beliefs, or formative events of a central character

Output

- A binary decision:
  - **1 (Consistent):** The backstory does not contradict the narrative
  - **0 (Contradictory):** At least one critical contradiction exists

Key Challenges

- Handling long-context narratives without truncation
- Avoiding surface-level keyword contradictions
- Reasoning over causality and character consistency across time

3. System Overview

We model the task as a hypothesis verification problem:

- **Novel → Ground Truth**
- **Backstory → Hypothesis**

The system operates through a structured pipeline:

1. Ingest the full novel
2. Chunk and index the text for efficient access
3. Decompose the backstory into atomic claims
4. Retrieve relevant narrative evidence for each claim
5. Evaluate each claim independently
6. Aggregate results deterministically into a final verdict

This modular design enables scalability and clear attribution of decisions.

4. Long-Context Handling and Chunking

To process the complete novel without loss of information, the text is divided into overlapping chunks. Each chunk represents a localized narrative unit, such as a paragraph group or short scene, and is assigned a unique identifier.

Benefits of Chunking

- Enables scalable processing of very long texts
- Allows focused evidence retrieval
- Preserves narrative continuity through overlap

This strategy ensures that no part of the novel is discarded while maintaining manageable reasoning units.

5. Pathway-Based Ingestion and Indexing

Pathway’s Python framework is used to ingest, structure, and index the chunked novel text. The system reads the novel from disk, converts it into chunked tables, and stores these chunks in a queryable format.

Key roles of Pathway in our system include:

- Streaming ingestion of large text files
- Structured representation of text chunks
- Efficient indexing for downstream retrieval

The indexed chunks serve as the evidence base for all subsequent reasoning steps.

6. Backstory Claim Decomposition

Rather than evaluating a backstory holistically, we decompose it into atomic, testable claims. Each claim represents a single assertion about the character, such as a belief, motivation, fear, or formative experience.

Claims are categorized as:

- **Core claims:** Fundamental to the character’s identity or long-term behavior
- **Non-core claims:** Secondary or contextual details

This decomposition avoids vague judgments and allows precise verification of each component of the backstory.

7. Evidence Retrieval Strategy

For each backstory claim, a structured textual query is constructed using key terms derived from the claim. This query is used to retrieve relevant narrative chunks from the indexed novel text.

The retrieval strategy is recall-focused, ensuring that potentially relevant evidence is not missed. Retrieved chunks may include:

- Character actions
- Dialogue
- Internal monologue
- Repeated behavioral patterns

By grounding each claim in multiple parts of the narrative, the system avoids reliance on isolated passages.

8. Claim-Level Reasoning and Aggregation

Each claim is evaluated independently against its retrieved evidence. The evaluation determines whether the claim is:

- Supported by the narrative
- Contradicted by the narrative
- Unconstrained (no clear evidence either way)

For Round 1, claim-level reasoning is demonstrated conceptually, with clear interfaces for automated LLM-based evaluation in future iterations.

Deterministic Aggregation Rule

- If **any core claim** is contradictory → Final decision = **0**
- Otherwise → Final decision = **1**

This rule prevents critical inconsistencies from being masked by majority voting.

9. Output Specification

The system outputs a CSV file (results.csv) with the following format:

story_id,prediction,rationale

story_XXX,0,A core backstory claim contradicts the narrative

story_YYY,1,All core backstory claims are consistent

This format ensures compatibility with automated evaluation pipelines.

10.Limitations and Future Work

Limitations

- Retrieval is currently based on structured textual matching rather than full semantic embeddings
- Claim-level reasoning is not fully automated in the prototype

Future Work

- Integrating semantic vector retrieval
- Automating claim evaluation using LLMs
- Extending the system to multi-character backstories

11.Conclusion

We present a clear, modular, and explainable system for verifying backstory–narrative consistency in long-form fiction. By combining chunk-based indexing, structured claim decomposition, and deterministic aggregation, the system provides reliable and interpretable decisions while handling long-context narratives effectively. The approach aligns closely with Track A’s emphasis on systems reasoning and provides a strong foundation for future extensions.