The 8-Second Constraint

One of the most technically demanding constraints of this project is the requirement that each generated scene corresponds to a video segment with a maximum duration of 8 seconds. This is not merely a file-size constraint but a narrative pacing constraint. It forces the system to break the flow of a novel into “micro-beats.”

The Temporal-Textual Conversion Problem

Text does not have an inherent duration. A single sentence (“The war lasted a hundred years.”) can imply a century, while three pages of stream-of-consciousness thought might occur in a split second. Converting text to time requires a heuristic algorithm based on **Speech Rate** and **Action Density**.

The “Time-Cost” Algorithm

We utilize a heuristic derived from voice-over and screenplay standards to estimate the duration of a text segment. The average speaking rate for clear, narrative voice-over is approximately 140 words per minute (wpm).

THE BASE CALCULATION

140 Words / 60 Sec = ~2.3 Words/Sec

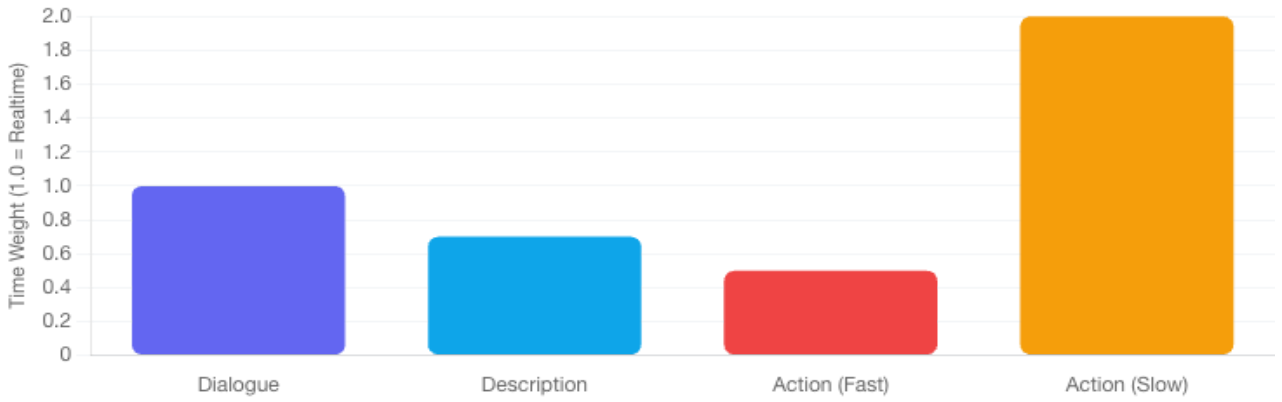
Target: ~18-20 Words per 8s Clip

However, a strict word count is insufficient because “action text” reads faster than “dialogue text.” Therefore, the **Chunking Agent** employs a weighted token analysis:

Token Type	Weight Multiplier	Rationale
Dialogue	1.0	Spoken at natural speed.
Descriptive	0.7	Visuals process faster than reading; a detailed description of a room can be shown in 2 seconds.
Action	Variable (0.5 - 2.0)	“He ran” (Fast/0.5). “He waited for the sun to set” (Time-lapse/2.0).

Weighted Token Analysis

Visualizing the time-cost multiplier for different content types.



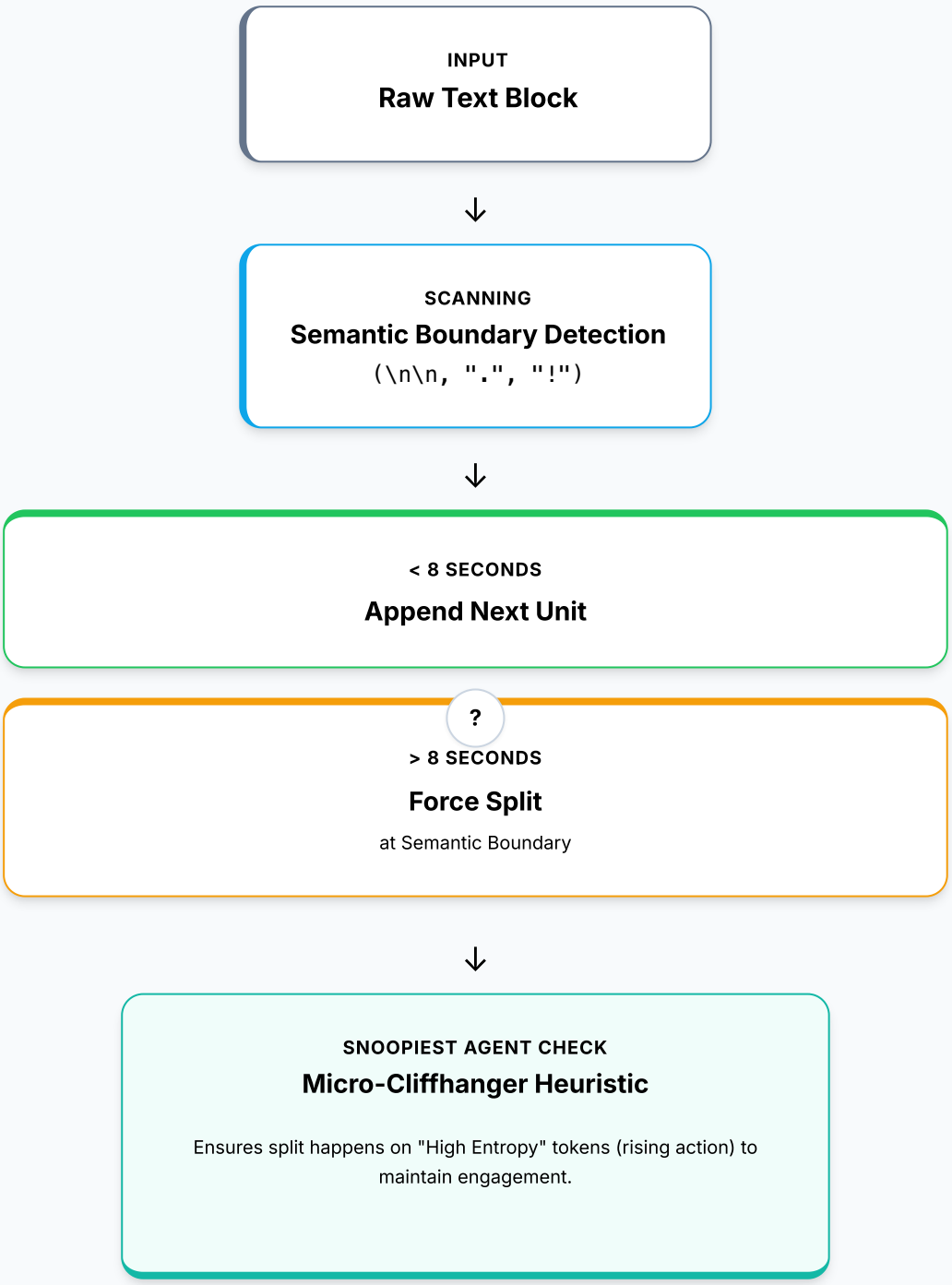
The Segmentation Architecture

The chunking process follows a strict “Atomic Scene” logic to ensure that video generation models (which often drift after 5-10 seconds) remain coherent.

1. **Input**: A block of text from the Markdown file.
2. **Semantic Boundary Detection**: The system scans for natural breaks like paragraph breaks (`\n\n`), dialogue tags (`"`, `”`), or sentence terminators (`.`, `?`, `!`).
3. **Duration Estimation**: The system calculates the estimated duration of the segment using the weighted algorithm.
    - *If Estimated Duration < 10s*: The system appends the next semantic unit.
    - *If Estimated Duration > 10s*: The system forces a split at the nearest semantic boundary (sentence or clause level).
4. **Coherence Check**: A lightweight “Snoopiest” sub-agent reviews the split. If a sentence is cut in a way that destroys meaning (e.g., split between subject and predicate), the boundary is shifted.

Segmentation Logic Flow

Decision tree for atomic scene generation



The “Micro-Cliffhanger” Heuristic

To maintain viewer engagement across these short 8-second clips, the chunking algorithm favors splits that end on “high entropy” tokens—words that imply unresolved action or rising intonation. This ensures flows naturally into the next clip.

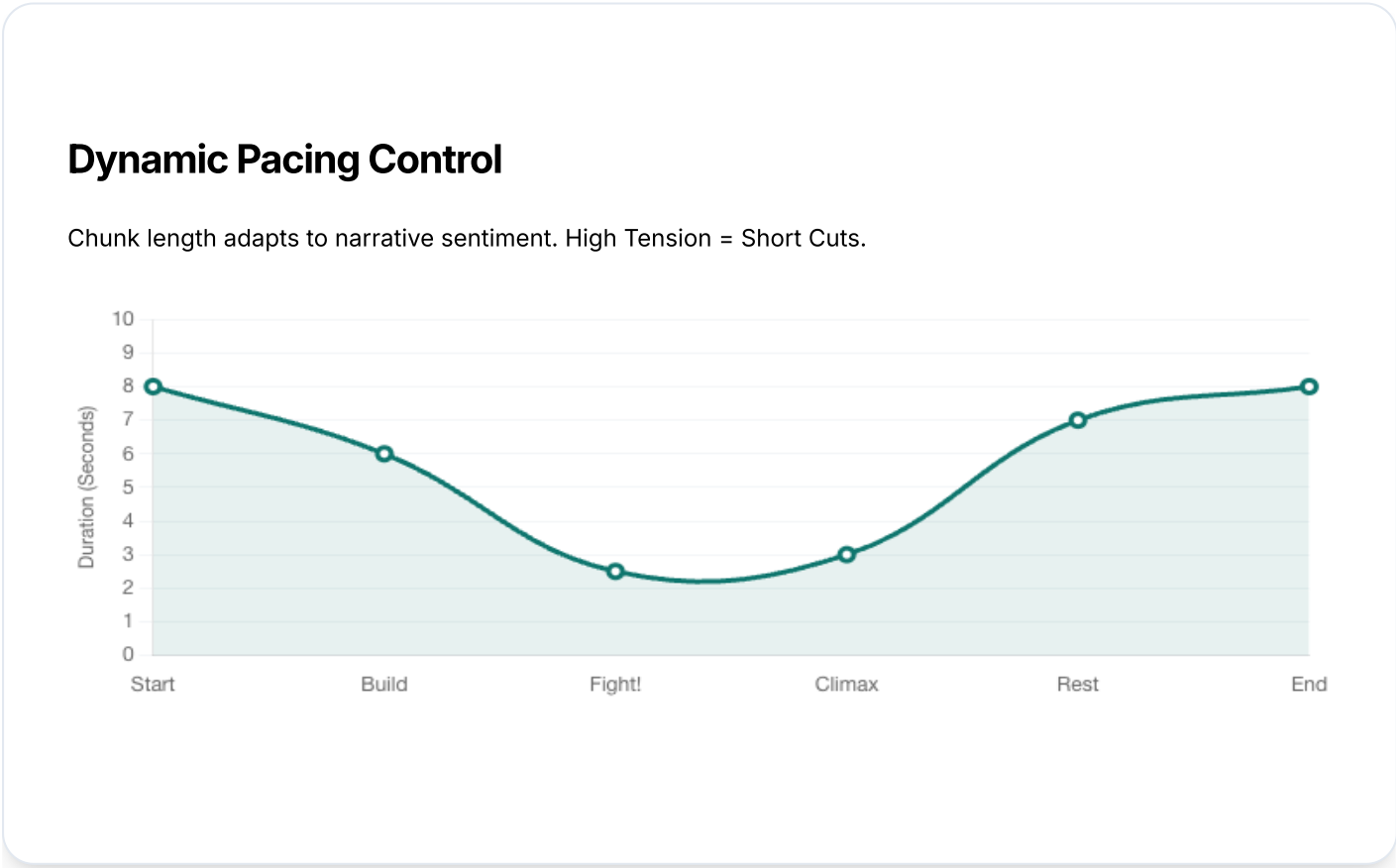
Deep Dive: The Chunking Heuristic and Pacing

The 8-second constraint is more than a technical limit; it is an aesthetic one. It dictates the “rhythm” of the generated video.

Dynamic Pacing Control

The Chunking Agent analyzes the **Sentiment and Pacing** of the text segment.

- **High Tension (Fight Scene)**: The agent intentionally creates shorter chunks (2-3 seconds). Even though the limit is 8s, rapid cuts increase tension.
- **Low Tension (Landscape Description)**: The agent maximizes the chunk to the full 8 seconds to allow for slow, panning camera movements.



The “Audio-Visual Split”

The architecture actually creates *two* parallel streams from the chunk:

1. **Visual Prompt**: (Used for Video Gen).
2. **Audio Prompt**: (Used for TTS/Audio Gen).

If the dialogue is too long, the system splits the Visual Prompt into two clips (Part A and Part B) but keeps the audio flowing across them.