

A SUPPLEMENTARY MATERIAL

A.1 RESULTS - README: BEFORE WATCHING COMPRESSED VIDEOS

Please refer to the supplementary video for more results.

README: BEFORE WATCHING COMPRESSED VIDEOS

Dear Reviewers,
Due to the **100MB submission limit** for ICLR 2026, **we had to significantly compress the supplementary files.** The original file size for the Main Results video was 179.3MB, and the Comparison video was 116MB. Please understand that the **compression version of videos may introduce visible artifacts, and we are happy to provide the original high-quality videos during the review process** if needed and approved by the conference.
Sincerely,
The Authors

A.2 ETHICS STATEMENT

Ethics Statement

All audio samples used in this work are limited to short excerpts for non-commercial, academic research purposes. No full scenes or monetized content are used, and speaker identities are simulated for character voice generation. We adhere to fair use guidelines and will release only anonymized and text-aligned metadata in compliance with copyright standards. We further clarify that all cartoon characters depicted in this research (Shrek & Donkey, Doraemon & Nobita, Tom & Jerry, and Minions Kevin & Bob) are used solely as fictional references to evaluate storytelling and voice synthesis capabilities. These characters are not used for profit, distribution, or endorsement, and their inclusion is intended for academic demonstration under fair use. Generated audio and visuals are produced synthetically and do not involve the use of any original footage or proprietary media. Our focus remains on advancing generative modeling techniques for educational and research purposes within responsible AI practices.

A.2.1 LICENSE FOR EXISTING ASSETS

License for existing assets

We utilize several publicly available pretrained models in our framework, each of which is used in accordance with its respective open-source license. Specifically, the Mochi-1 video generation model is distributed under the Apache 2.0 License (<https://github.com/genmoai/mochi>), and is used for scene-level visual synthesis. The Sesame Conversational Speech Model (CSM), used for character-specific expressive speech generation, is also distributed under the Apache 2.0 License (<https://huggingface.co/sesame/csm-1b>). In addition, the BLIP model, which is distributed under the Creative Commons Attribution 4.0 International (CC BY 4.0) License (<https://huggingface.co/Salesforce/blip-image-captioning-base>). All models are used without modification and solely for academic, non-commercial research purposes. We ensure proper attribution and full compliance with each model’s licensing terms.

A.3 USER STUDY

Survey Setup and Interface. Participants were presented with 15 comparison questions, each showcasing 2–4 short AI-generated video clips (labeled only as “Video A”, “Video B”, etc. to ensure fairness). Each video was accompanied by synthesized character speech and dialogue. The clips varied across different experimental settings including speech synthesis methods, dialogue generation approaches, and ablation configurations.

After watching each video set, participants were asked to select the most vivid and engaging clip and explain their reasoning. They could either choose from a list of predefined qualitative factors or write open-ended comments. The predefined options included criteria such as:

- *Because the dialogue felt natural and matched the scene*
- *Because the character’s tone and personality were well expressed*
- *Because the character’s actions and dialogue were well aligned*
- *Because the background visuals or movements were natural and immersive*
- *Because the context between scenes was connected, making the story flow smoothly*
- *Because the voice conveyed rich emotions and suited the character*
- *Because the combination of scene and dialogue was intuitively understandable*
- *Because the character’s voice was consistent and felt familiar*
- *Because overall, it felt immersive and vivid*
- *Because each scene provided new information without repetitive expressions*
- Participants could also submit free-form feedback if none of the listed items captured their impression.

Conditions. The 15 questions were organized into five experimental categories:

- **Speech Generation & Speaker Conditioning (Q1–Q3):** Comparison across our full model, versions without character embeddings, and baseline systems like Bark and Tacotron.
- **Dialogue Generation (Q4–Q6):** Our system versus a BLIP-based caption-to-dialogue baseline.
- **End-to-End Comparison (Q7–Q9):** Full pipeline compared to prior works like Mochi and Vlogger.
- **Recursive Narrative Bank (Q10–Q12):** Ablation of our Recursive Narrative Bank (RNB).
- **Key Frame Image Conditioning (Q13–Q15):** Ablation of visual grounding in dialogue generation.

Protocol. The survey took approximately 15–20 minutes. Participants were advised to use headphones for best audio quality but could participate via any device supporting video and audio. To respect IRB policy, no personal data was collected, and we do not compensate the participants. The study was reviewed and exempted by the Institutional Review Board (IRB).

Statistical Summary. Out of **225** total responses, we observed consistently strong preference for our full model across all experiment categories. Rather than aggregating across all videos (which may appear in different subsets of questions), we report per-condition confidence intervals for Ours only. This avoids bias from unequal appearance of baseline systems. Table 2 summarizes the proportion of Ours selections and their 95% confidence intervals across the five experiment types. Also, this table further details per-condition confidence intervals for Ours. In all five experimental categories, our method was selected by a clear majority, with proportions ranging from 84.4% to 93.3%.

Experiment	Ours Proportion	95% CI Lower	95% CI Upper	Total Votes
Speech Generation	0.9333	0.8605	1.0000	45
Dialogue Generation	0.8444	0.7386	0.9503	45
End-to-End Comparison	0.9111	0.8280	0.9943	45
Recursive Narrative Bank	0.8444	0.7386	0.9503	45
Key Frame Conditioning	0.8444	0.7386	0.9503	45

Table 2: **Per-experiment condition preference for Video A with 95% confidence intervals.** Our method was selected by a clear majority, with proportion from 84.4% to 93.3% in all five categories.

Reasoning Analysis. Participants justified their selections using a list of qualitative reasons. Table 3 shows the distribution of all justifications. The most frequently selected reason was “*Because the dialogue felt natural and matched the scene*” (42.2%), followed by “*Character tone and personality*” (21.3%). Free-form responses that did not match predefined categories were grouped as “Other” (1.3%). Note that categories with fewer than 5 responses yielded wider confidence intervals with potentially negative lower bounds due to normal approximation, but these are clipped at zero in our interpretation.

Reason Category	Proportion	95% CI Lower	95% CI Upper
Dialogue naturalness	0.4222	0.3577	0.4868
Character tone/personality	0.2133	0.1598	0.2669
Action-dialogue alignment	0.0756	0.0410	0.1101
Visual immersion	0.0622	0.0307	0.0938
Scene-to-scene continuity	0.0578	0.0273	0.0883
Emotional voice	0.0533	0.0240	0.0827
Scene-dialogue coherence	0.0444	0.0175	0.0714
Consistent voice	0.0311	0.0084	0.0538
Overall immersion	0.0178	0.0005	0.0350
Other	0.0133	0.0000	0.0283
Content novelty	0.0089	0.0000	0.0212

Table 3: **Participant justifications for their preference with 95% CI:** the most frequently selected reason was “the dialogue felt natural and matching the scene”.

A.4 DETAILED QUANTITATIVE EVALUATION & ABLATION

Due to a formatting error, please check the Datasets in Appendix 6, User Study in Appendix 3. We will revise that in the camera-ready version.

Detailed Quantitative Evaluation To evaluate *dialogue quality*, we compute BERTScore and BLEU between generated utterances and their paired scene prompts and image-grounded captions. As summarized in Table 4, our full pipeline achieves the highest scores on both metrics (BERTScore: 0.0674 ± 0.057 , BLEU: 1.8726 ± 2.7237), indicating strong semantic fidelity and fluency. Ablating visual grounding (i.e., “Without KeyFrame”) degrades BERTScore to 0.0392 ± 0.1187 and BLEU to 0.8609 ± 1.1529 , with large variances suggesting unstable and inconsistent generation. Similarly, removing the Recursive Narrative Bank (RNB) drops BERTScore to 0.0094 ± 0.0353 and BLEU to 0.5268 ± 0.6288 , underscoring the RNB’s role in linguistic and narrative coherence.

For *multimodal alignment*, we report CLIPScore between generated dialogue and keyframe images. Our model yields a CLIPScore of 27.4822 ± 3.4597 , outperforming alternative pipelines such as Vlogger+Speech (26.1703 ± 4.0179) and Mochi+Speech (26.6107 ± 4.9015). Notably, removing keyframe conditioning drops CLIPScore to 25.3061 ± 3.2293 , indicating weaker alignment with visual context.

To measure *speech expressivity*, we apply Dynamic Time Warping (DTW) over pitch contours to compare generated speech with reference character audio. Our system achieves the lowest DTW value (16.2412 ± 5.7724), indicating strong temporal and prosodic alignment. Ablating speaker rendering increases DTW substantially to 49.6302 ± 55.1206 , with the large standard deviation highlighting inconsistent prosody. Comparisons with Bark (16.4668 ± 5.3581) and Tacotron 2 (16.4484 ± 0.9273) show that our method achieves slightly better average performance while maintaining competitive expressivity with lower variance.

These results are summarized in Table 4. The consistent inclusion of standard deviations across all methods allows us to assess both performance and stability. Collectively, these metrics and their statistical spread confirm that each module—visual grounding, dialogue memory, and speech conditioning—contributes meaningfully and measurably to the multimodal storytelling quality.

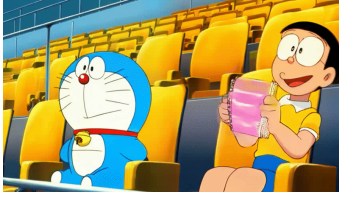
Detailed Ablation Study We first examine the effect of keyframe-based visual grounding in dialogue generation. When image features are removed (“w/o Conditioning”), performance drops sharply across all metrics: BERTScore falls from 0.0674 to 0.0392, BLEU from 1.8726 to 1.2914,

Method	BERTScore	BLEU	CLIP	DTW
Ours	0.0674 \pm 0.057	1.8726 \pm 2.7237	27.4822 \pm 3.4597	16.2412 \pm 5.7724
BLIP	0.0674 \pm 0.057	1.8726 \pm 2.7237	27.4822 \pm 3.4597	17.0377 \pm 7.5249
Speech w/o rendering	0.0674 \pm 0.057	1.8726 \pm 2.7237	27.4822 \pm 3.4597	49.6302 \pm 55.1206
Bark	0.0674 \pm 0.057	1.8726 \pm 2.7237	27.4822 \pm 3.4597	16.4668 \pm 5.3581
Tacotron 2	0.0674 \pm 0.057	1.8726 \pm 2.7237	27.4822 \pm 3.4597	16.4484 \pm 0.9273
Mochi	0.0092 \pm 0.0474	0.1272 \pm 0.1734	26.6107 \pm 4.9015	17.7431 \pm 4.8291
Vlogger	0.0094 \pm 0.0842	0.2648 \pm 0.6927	26.1703 \pm 4.0179	18.6999 \pm 7.7477
Without RNB	0.0094 \pm 0.0353	0.5268 \pm 0.6288	27.2259 \pm 3.8112	16.5604 \pm 5.1673
Without KeyFrame	0.0392 \pm 0.1187	0.8609 \pm 1.1529	25.3061 \pm 3.2293	22.4406 \pm 19.7112

Table 4: **Quantitative Evaluation with Mean \pm Standard Deviation.** Our method achieves the best or competitive performance across linguistic, multimodal, and speech metrics, with standard deviations demonstrating consistent reliability.

P1: They find seats with a good view of the stands and set down their gear

P2: Doraemon and Nobita are sitting in the press area, adjusting their notepads



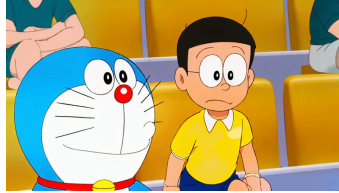
Ours Doraemon: Of course the Goal-in-ator!

w/o RNB Doraemon: Are you ready to doodle on that notepad?

w/o KFC Nobita: We could report on an ant's game plan!

P1: Doraemon and Nobita watch players lining up for kickoff sitting in the stands

P2: Doraemon and Nobita are focusing intently, eyes fixed on the field



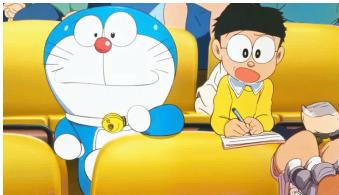
Ours Doraemon: Just aim and BAM goal every time!

w/o RNB Nobita: Are they gonna score in every game Doraemon?

w/o KFC Nobita: I can even see the coach's nose hair strategy!

P1: As the match begins, Nobita jots quick notes while Doraemon scans the field with binoculars sitting in the seats

P2: Doraemon and Nobita are sitting and concentrating on the game



Ours Nobita: Can I borrow the Goal-in-ator?

w/o RNB Nobita: That also predict the snacks they will serve later?

w/o KFC Nobita: Like a tangled spaghetti strategy!

Figure 5: **Qualitative Ablation Results on Dialogue Coherence. (Full Version)** We showcase dialogue generated by our full model compared to ablated variants without the Recursive Narrative Bank (w/o RNB) and without Keyframe Conditioning (w/o KFC). The absence of RNB leads to disrupted narrative flow across scenes, while removing KFC yields contextually irrelevant or less grounded utterances. These results highlight the importance of both modules in producing coherent, character-consistent dialogue.

CLIPScore from 27.4822 to 25.3061, and DTW increases from 16.2412 to 22.4406. These declines confirm that grounding utterances in visual context is critical for generating coherent, character-consistent dialogue that aligns with the scene.

Next, we evaluate the impact of speaker conditioning in speech synthesis. Removing reference-based character embeddings leads to a significant rise in DTW—from 16.2412 to 49.6302—indicating a loss of prosodic consistency and character identity. Comparisons with other synthesis models such as Bark (16.4668) and Tacotron 2 (16.4484) show that while these methods offer reason-

able performance, our reference-guided approach produces more expressive and character-aligned speech.

We also conduct an end-to-end comparison by plugging our dialogue and speech components into different video generation frameworks. When combined with Mochi or Vlogger, overall quality degrades noticeably: BERTScore drops below 0.01, BLEU falls below 0.3, and CLIPScore and DTW also worsen. These results suggest that despite strong language and audio generation, weaker video backbones limit overall storytelling quality. This demonstrates the importance of tight integration between motion modeling and narrative components.

To assess narrative coherence, we ablate the Recursive Narrative Bank (RNB), which maintains a memory of prior scenes and utterances. Without RNB, BERTScore declines to 0.0094 and BLEU to 0.5268, even though CLIPScore and DTW remain stable. This indicates that RNB primarily enhances linguistic continuity and long-range consistency in dialogue, which are essential for maintaining character development across scenes.

In summary, each component—visual grounding, speaker conditioning, and narrative memory—plays a distinct and complementary role in the system. The qualitative differences in Figure 5, along with quantitative improvements across all metrics, underscore the effectiveness of our integrated design.

A.5 RECURSIVE NARRATIVE BANK AND SCRIPT THEORY

The Recursive Narrative Bank (RNB) is designed to enable coherent, character-driven dialogue generation in long-form multimodal storytelling by drawing on structured representations of memory inspired by cognitive science. Rather than functioning as a traditional memory buffer that passively recalls past utterances, RNB serves as a role-aware, temporally recursive scaffolding mechanism that simulates human-like conversational behavior over time. This distinction is essential: what makes dialogue compelling in extended narratives is not the surface-level repetition of past phrases, but the ability to produce behaviorally and emotionally consistent responses that evolve with the scene, role, and intention.

This principle is grounded in Script Theory (Schank & Abelson (2013); Bower et al. (1979); Wilensky (1983)), a foundational framework in cognitive psychology that models how humans interpret and produce behavior in structured situations. According to Script Theory, people rely on internalized “scripts”—sequences of stereotyped events and role expectations—to understand what to say, when to say it, and how to act in social contexts. For instance, a script for dining in a restaurant involves roles (customer, waiter), actions (ordering, serving), and expectations (e.g., the bill comes after dessert, not before). Dialogue within such scripts is not retrieved verbatim but generated based on the evolving context and the agent’s role within it. The recursive structure of RNB operationalizes this by maintaining separate, character-specific dialogue histories that are re-injected into prompt templates in a way that mirrors how scripts are mentally activated and updated by humans during interaction.

Each RNB prompt is thus not a flat history window, but a structured invocation of a script fragment: it conditions the current scene on the immediate visual context (e.g., keyframe and action prompt) and the speaker’s evolving narrative state. The inclusion of scene-level grounding ensures that the model’s responses are not just temporally coherent but also visually relevant, maintaining alignment between what is said and what is shown. Moreover, by explicitly maintaining separate narrative banks for each character, the system supports differentiated behavioral trajectories—allowing one character’s tone to escalate while another’s remains calm, consistent with how individuals behave differently within the same scene.

This approach stands in contrast to generic prompt memory methods that treat history as unordered or speaker-agnostic. RNB is structured around the core dimensions of script-based modeling: temporality, role-awareness, and goal-consistent progression. Its recursive update mechanism ensures that the memory bank evolves over time without growing unbounded, allowing dynamic narrative control while remaining compatible with stateless large language model APIs. This makes RNB particularly well-suited to real-world generative systems, where persistent fine-tuned memory is unavailable but coherence remains critical.

By explicitly modeling narrative evolution through recursive, structured, character-aware prompts grounded in visual context, RNB enables zero-shot dialogue generation that reflects how humans produce situated language—not from scratch, but from structured expectations embedded in unfolding events. This theoretical grounding offers not only interpretability and generalizability but also a cognitively motivated lens for designing narrative-capable generative architectures.

To enhance clarity, we provide a formalized description of the end-to-end process used to generate character-consistent dialogue from scene prompts and visual context.

Let $(p_1^{(t)}, p_2^{(t)})$ denote the structured prompt pair at scene timestep t . These are concatenated to form the full prompt:

$$p^{(t)} = p_1^{(t)} + \text{``. ``} + p_2^{(t)}. \quad (7)$$

Given the generated video clip $x_v^{(t)}$ for this prompt pair, we extract the middle frame:

$$I^{(t)} = \text{SampleFrame}(x_v^{(t)}, t = T/2), \quad (8)$$

where T is the total number of frames.

We then compute a high-level visual semantic representation:

$$c^{(t)} = \text{SceneFeat}(I^{(t)}), \quad (9)$$

where `SceneFeat` is implemented using a pretrained BLIP captioning model. The result is a natural language caption aligned with the scene’s visual content.

To ensure continuity across scenes, we define a Recursive Narrative Bank \mathcal{H}_t as a temporally recursive memory:

$$\mathcal{H}_t = \{x_d^{(t-1)}, x_d^{(t-2)}, \dots, x_d^{(t-N)}\}, \quad (10)$$

where each $x_d^{(i)}$ is a dialogue utterance generated at timestep i , and N defines the memory window (e.g., $N=\text{all}$ in our implementation).

This memory, along with the current scene and visual embedding, is embedded into a structured input for the language model:

$$\text{Input}^{(t)} = [\text{Scene}] p^{(t)} \parallel [\text{Image}] c^{(t)} \parallel [\text{DialogueMemory}] \mathcal{H}_t. \quad (11)$$

The character-specific dialogue for the current scene is then generated via:

$$x_d^{(t)} = \text{LLM}(\text{Input}^{(t)}), \quad (12)$$

where `LLM` is a pretrained stateless large language model (e.g., GPT-4o). Only a single character speaks per turn, and speaker role is determined externally by prompt scheduling.

Finally, the narrative bank is updated recursively:

$$\mathcal{H}_{t+1} = \text{Truncate}(\mathcal{H}_t \cup \{x_d^{(t)}\}), \quad (13)$$

where `Truncate` enforces the memory limit N by removing the oldest entry if necessary.

Cognitive Perspective. From the perspective of Script Theory [Schank & Abelson (2013); Bower et al. (1979); Wilensky (1983)], each structured prompt $\text{Input}^{(t)}$ simulates a localized fragment of a behavioral script. Rather than relying on rote memory, the model is guided by structured expectations derived from evolving visual and narrative cues. The separation into [Scene], [Image], and [DialogueMemory] reflects the key components of human-scripted interaction: situational setting, perceptual input, and role-consistent behavioral priors. This decomposition enables zero-shot stateless generation while preserving coherent narrative flow.

A.6 CONVERSATIONAL SPEECH GENERATION WITH RESIDUAL VECTOR QUANTIZATION

Conventional text-to-speech models directly map textual input to audio but often fail to reproduce the variability of conversational prosody. To overcome this limitation, we follow a residual vector quantization (RVQ) framework that represents continuous waveforms as discrete tokens, enabling

transformer-based modeling of both text and audio in a shared space [AI \(2024b\)](#). Two types of tokens are used: *semantic tokens*, which encode phonetic and linguistic content in a speaker-invariant manner but act as a prosodic bottleneck, and *acoustic tokens*, which preserve fine-grained attributes such as timbre, identity, and rhythm via RVQ. While semantic tokens provide a compact high-level abstraction, acoustic tokens are crucial for reconstructing high-fidelity and natural-sounding speech.

Let the text sequence be $T = \{t_1, \dots, t_n\}$ and the conversational history be $A = \{a_1, \dots, a_m\}$. The backbone transformer autoregressively models the zeroth-level codebook k_0 as

$$p(k_0 | T, A) = \prod_{t=1}^{n+m} p(k_{0,t} | k_{0,<t}, T_{\leq t}, A_{\leq t}), \quad (14)$$

where k_0 captures semantic and coarse prosodic structure. A lightweight decoder then reconstructs the higher-level residual codebooks $\{k_1, \dots, k_{N-1}\}$ conditioned on k_0 :

$$p(k_{1:N-1} | k_0) = \prod_{i=1}^{N-1} \prod_t p(k_{i,t} | k_{i,<t}, k_{<i}, k_0). \quad (15)$$

Here, each level k_i refines acoustic resolution by conditioning on both its own history and all lower-level codebooks. Because RVQ imposes sequential dependence across levels, a *delay-pattern scheme* is employed in which higher codebooks are temporally offset to ensure conditioning on lower-level predictions. This improves fidelity but increases the time-to-first-audio, scaling linearly with the number of codebooks N .

To balance expressivity and efficiency, the Conversational Speech Model (CSM) separates modeling into two parts: a multimodal backbone for k_0 and a smaller decoder for $\{k_1, \dots, k_{N-1}\}$. Generated acoustic tokens are autoregressively fed back into the backbone until an end-of-token symbol is reached, yielding coherent conversational speech. Text tokens are produced via a LLaMA tokenizer, and audio tokens are derived from a split-RVQ tokenizer that outputs one semantic and $N - 1$ acoustic codebooks at 12.5 Hz.

Training such models presents a severe computational burden because the effective batch size is $B \times S \times N$, where B is the batch size, S the sequence length, and N the number of RVQ levels. To mitigate this, we adopt a *compute amortization* strategy in which the backbone is trained on all frames for k_0 , while the decoder is updated only on a random subset $\mathcal{F}' \subset \mathcal{F}$ of frames (with $|\mathcal{F}'|/|\mathcal{F}| = 1/16$). The loss function is thus

$$\mathcal{L} = \sum_{f \in \mathcal{F}} \mathcal{L}_{k_0}(f) + \sum_{f \in \mathcal{F}'} \sum_{i=1}^{N-1} \mathcal{L}_{k_i}(f), \quad (16)$$

where $\mathcal{L}_{k_i}(f)$ is the cross-entropy loss for predicting codebook k_i at frame f . Empirically, this amortized scheme reduces memory and training cost without degrading perceptual quality.

Finally, to evaluate contextual speech generation, we employ both objective and subjective measures. Objective metrics include word error rate (WER) and speaker similarity (SIM), which saturate at near-human performance, as well as newly introduced benchmarks such as homograph disambiguation (e.g., distinguishing *lead* /lɛd/ vs. /li:d/) and pronunciation consistency across multi-turn speech. Subjective evaluation is conducted via Comparative Mean Opinion Score (CMOS) studies, where listeners compare model outputs against human recordings both with and without conversational context. While results show no significant difference without context, humans are consistently preferred when context is included, indicating that conversational prosody remains an open challenge. Limitations also remain in language coverage (primarily English) and the inability to fully capture higher-level turn-taking structures, though scaling model size and dataset diversity shows consistent improvement.

A.7 COMPUTATION TIME AND MEMORY CONSUMPTION

Table 5 provides a breakdown of computation time and GPU memory consumption for our full system and its subcomponents. The full pipeline consumes approximately 31.21 GB of memory and takes 869 seconds to process a batch of stories, reflecting the combined cost of visual grounding, dialogue generation, and expressive speech synthesis.

To better understand individual module efficiency, we separately measure:

Module	Step	Memory Consumption	Computation Time
Text2Story Kang et al. (2025)	Inference	31.21 GB / 80.0 GB (29,767 MiB)	869 sec
Natural Language Module (NLM)	Inference	0.00 GB / 80.0 GB (0 MiB)	6.46 sec / story (avg)
Speech Synthesis	Inference	4.78 GB / 80.0 GB (4,567 MiB)	4.04 sec / story (avg)

Table 5: **Computation Time and Memory Consumption Analysis.** We report detailed computation statistics of the full model and its key submodules during inference, evaluated on an NVIDIA H100 GPU (80GB). The full pipeline integrates vision-language grounding, narrative modeling, and speech generation.

- **Natural Language Module (NLM)** – responsible for character-consistent dialogue generation based on prompt pairs and visual features. It consumes negligible GPU memory (0 MiB) and takes **6.4552 seconds per story on average**, totaling **83.92 seconds** for a representative case (Shrek & Donkey in San Francisco).
- **Speech Synthesis** – performed using a reference-guided speech generation model. It consumes **4.57 GB GPU memory** and requires **4.0389 seconds per story on average**, totaling **59.62 seconds** for the same case.

Although one may question whether lighter solutions like Mochi or Vlogger are preferable given their faster inference time (e.g., 126 sec total for Mochi), such comparisons overlook the quality-performance trade-off. Our full model is specifically optimized for coherent narrative flow, persona-consistent dialogue, and emotionally expressive speech. We justify the computational overhead through end-to-end evaluation, where our system significantly outperforms ablated or simplified baselines across automated and human preference metrics (see Table 1, Figure 4).

In summary, our pipeline demonstrates a strong balance between memory efficiency, inference time, and qualitative output. This makes it not only scalable but also effective for real-world storytelling applications where coherence, fidelity, and expressivity are essential.

A.8 DATASETS (VIDEO GENERATION, AUDIO GENERATION)

We provide a structured benchmark for multimodal story generation across diverse narrative settings using familiar animated characters. This benchmark consists of multiple scene-level video clips, each represented by paired prompts: one describing the scene (*Prompt 1*) and another specifying the action (*Prompt 2*). Our benchmark is designed to test both the visual and auditory fidelity of generated narratives in character-driven storytelling.

Video Generation. Our video generation dataset includes four themed narrative scenarios: **Urban Exploration in San Francisco**, **Nightlife in Las Vegas**, **Outdoor Cooking Show**, and **Sports Reporting Commentary**, all featuring familiar animated character pairs such as *Shrek & Donkey*, *Doraemon & Nobita*, and others. Each story comprises 11–13 sequential scene-action pairs (22–26 prompts in total), reflecting smooth scene transitions and character continuity. The prompts describe everyday actions (e.g., walking, sitting, looking) as well as location-specific interactions (e.g., cooking, cheering, jogging), making them well-suited for evaluating narrative coherence and multimodal alignment.

The **San Francisco** sequence begins with two characters arriving at the airport, retrieving their suitcase, and exploring iconic city landmarks including the Painted Ladies, Palace of Fine Arts, and the Golden Gate Bridge. The narrative concludes with a tranquil moment at Battery Spencer during sunset. The prompts highlight contextual changes in transportation (airplane, SUV, cable car), movement (walking, jogging), and visual engagement (gazing, admiring the view), facilitating fine-grained temporal generation.

In contrast, the **Las Vegas** narrative focuses on nighttime entertainment and visual spectacle. Starting with a walk along the Las Vegas Strip, two characters experience the Bellagio Fountain, interact with a slot machine, and later visit Fremont Street. This scenario emphasizes vivid lighting conditions, expressive reactions, and physical interactions, which are critical for evaluating temporal coherence and spatial attention in video generation.

The **Outdoor Cooking Show** features two characters at a forest campsite as they go through the process of preparing hot dogs. The narrative includes gathering ingredients, lighting a fire, and enjoying the meal. This instructional and grounded setup enables the assessment of stepwise procedural generation and causal alignment between actions and objects.

Finally, the **Sports Reporting** scenario depicts characters acting as soccer commentators in a stadium. Two characters provide play-by-play analysis, react to match events. This setting demands precise modeling of conversational rhythm, character roles, and referential language grounded in visual context, testing the system’s ability to maintain long-term speaker consistency and context awareness.

Prompt Format. Each scene is defined by a prompt pair (p_1, p_2) , where p_1 establishes the setting and p_2 specifies the action. For example:

- prompt_san_francisco_shrek_V_2122:
[*"The sun begins to set over the Pacific Ocean", "Shrek and Donkey are standing"*]
- prompt_vegas_shrek_V_1314:
[*"Shrek and Donkey press a button on a slot machine in Las Vegas at night", "Shrek and Donkey are sitting"*]

Each full narrative includes 11–13 such prompt pairs (per setting), resulting in 22–26 scene-specific inputs per video. These are used to condition both video diffusion and dialogue generation models.

Audio Generation. To generate character-consistent speech, we use short voice clips publicly available on YouTube, released by official movie or studio channels. Specifically, our dataset includes two audio samples representative of expressive and emotionally charged dialogue by Shrek and Donkey, respectively. These clips serve as reference prompts for synthesizing conversational speech across scenes.

- conversational_a (Shrek, 10sec):
"We? Donkey, there's no we. There's no our. There's just me and my swamp. And the first thing I'm gonna do is build a 10-foot wall around my land."
- conversational_b (Donkey, 10sec):
"Yes, I was talking to you. Can I just tell you that you was really great back there, man? Those guards thought they was all that. Then you showed up and bam!"
- conversational_a (Doraemon, 8sec):
"Yep, first, the materials. Do you have any plastic lying around? Dump them in the mecha-maker."
- conversational_b (Nobita, 9sec):
"We can make a ship with this thing? What do you mean? I got all these old toys I don't play with anymore."
- conversational_a (Tom, 12sec):
"I'm Tom. You think I'm a dummy? Hey you little pipsqueak! How come you never spoke before!"
- conversational_b (Jerry, 8sec):
"I'm Jerry. You said it, I didn't. There was nothing I wanted to say that I thought you'd understand."
- conversational_a (Minions Kevin, 5s):
"Okay Doamato rapita ra polka moba ratriba findoreba bas."
- conversational_b (Minions Bob, 9sec):
"Hm, uh, okay, okay, rakika, rebibas, Tony, prato, Tom, usaka, decrease, puratino."

These reference clips are extracted from well-known scenes in the *Shrek* franchise and other iconic series and are used strictly for academic and non-commercial research. The usage complies with the United States **Fair Use Doctrine**, which permits limited use of copyrighted material for purposes such as research, teaching, and scholarship. We ensure that:

- The **reference clips** used as inputs are short excerpts (each under 30 seconds) taken from publicly available YouTube videos released by official sources. These are not redistributed or reused directly in our outputs, and they are not used in a manner that competes with the original work.
- These clips are used solely to **guide the prosody and expression** of our speech synthesis system. The **final generated audio** is newly synthesized and does not contain or replicate the original audio segments.
- The synthesized voices approximate character tone and emotion but avoid reproducing identifiable voiceprints or actor likenesses, thus minimizing ethical and legal concerns.

Our generated audio is an expressive approximation, not an imitation. It preserves the **persona and rhythm** of the character while avoiding the reproduction of identifiable voiceprints. This approach minimizes ethical and legal concerns while enabling consistent, role-aware voice synthesis across narrative scenes.

A.9 ADDITIONAL ANALYSES: VOICE CONSISTENCY, PERSONA MODELING, AND ROBUSTNESS

Voice Consistency Across Characters. To further address reviewer concerns regarding occasional inconsistencies in generated speech, particularly for characters with monotonic delivery and low lexical diversity, we present a quantitative analysis of the reference voice prompts. Table 6 summarizes key lexical and acoustic properties across a diverse set of character types.

Character	Unique Words	Voiced Phonemes	Pitch Std (Hz)	Pause Ratio	Duration (sec)
Shrek	23	70	57.77	1.00	10.00
Donkey	38	118	83.79	1.00	10.00
Doraemon	23	70	106.73	0.98	8.37
Nobita	24	59	146.51	0.94	9.81
Kevin	9	33	79.80	1.00	5.24
Bob	11	40	139.87	1.00	9.47
Tom	28	73	127.81	1.00	12.68
Jerry	18	57	136.64	1.00	8.16

Table 6: **Reference Prompt Statistics.** Voice prompts differ significantly in lexical richness, prosodic dynamics, and temporal structure. High pitch variability does not necessarily correlate with expressive or consistent speech style; utterances with low lexical variety and near-continuous voicing (pause ratio ≈ 1.0) tend to sound more monotonic. This suggests that consistent speech perception depends on the interplay between lexical diversity, phoneme variation, and rhythmic structure—not pitch alone.

Voice Consistency Across Characters: Metric Descriptions. Each column in Table 6 is computed to quantify the linguistic and prosodic variability of the reference voice prompts:

- **Unique Words:** The number of distinct alphabetic tokens in the transcript, obtained using the NLTK tokenizer. This reflects lexical diversity, which contributes to the perceived richness and individuality of a character’s speech.
- **Voiced Phonemes:** The number of voiced phonemes (e.g., /b/, /m/, /a/) in the transcript, extracted using the `phonemizer` library with the `espeak` backend (en-us). This metric captures phonetic variety and articulation complexity.
- **Pitch Std (Hz):** The standard deviation of the estimated fundamental frequency (F0), computed using WORLD vocoder methods (`pyworld.harvest` and `stonemask`). Higher values indicate greater prosodic variation and expressive tone.
- **Pause Ratio:** The proportion of silent frames in the audio, based on short-time energy computed with a sliding window (frame length = 2048, threshold = 0.01). A ratio near 1.0 indicates continuous voicing with few pauses, often perceived as monotonic delivery.
- **Duration (sec):** Total length of the prompt audio, calculated as the number of samples divided by the sampling rate.

These metrics jointly characterize the structure and expressivity of each prompt. We observe that characters with high pitch variance but low lexical diversity and minimal pauses (pause ratio ≈ 1.0) tend to exhibit more frequent identity drift over long conversations. This suggests that speaker consistency is influenced not just by acoustic features, but also by linguistic variety and temporal structure. Future work may explore multi-turn prosody conditioning or speaker-aware memory to improve stability for such monotonic speech patterns.

Character Persona Modeling. Our system does not rely on static personality templates or trainable embeddings. Instead, character traits are dynamically inferred from the interplay of:

- the scene and action prompts,
- keyframe image captions (BLIP),
- and recursively accumulated dialogue history (via RNB).

This flexible mechanism allows for emergent behaviors and contextually appropriate responses, even for unseen or sparsely referenced characters like Tom and Jerry—who, as you may recall, rarely speak at all. In fact, their only widely known speaking moment comes from the delightfully controversial 1992 film *Tom and Jerry: The Movie*, which we bravely use as our sole reference. Rather than a limitation, we consider this an extreme zero-shot challenge: can the system generate character-faithful dialogue for two icons of silence? While the movie may have divided fans, our pipeline passed the test—producing speech that’s chaotic, emotionally erratic, and somehow still on-brand. Just like Tom and Jerry themselves.

Memory Length and Long-form Generation. The Recursive Narrative Bank (RNB) retains *all* prior utterances by default ($N = \text{all}$), ensuring long-range dependency modeling and character continuity. Each dialogue turn remains compact (less than 100 characters), resulting in manageable prompt lengths. We have successfully generated stories with over 13 prior utterances without encountering token limits or degradation. This design supports scalable storytelling over extended sequences without truncation or loss of coherence.

Robustness to Model Substitution. Our modular pipeline decouples each stage (video, dialogue, speech) using intermediate natural-language representations. When we substitute the visual module (e.g., replacing Text2Story with Mochi or Vlogger), downstream components remain stable. As illustrated in Figure 5, quality degrades only locally (e.g., visual-semantic alignment) without cascading failures. This architecture supports component-level improvements and domain transfer without retraining the full pipeline.