Automatic Narration of Movies via NarrativeML[*]

Inderjeet Mani

Atelier Milford Paradise, Moo 1 Paknampran, Pranburi, Prachuap Kiri Khan, 77220 Thailand

Abstract

Automatic narration of movies can be useful for the visually impaired and for adding additional dimensions of experience to silent films. Recent Generative AI advances have made it possible to describe sequences of images in natural language. However, the depth of AI understanding of film narrative can be unclear. To remedy this, the AI can auto-annotate the narrative content of the video in terms of an expressive narrative annotation language called NarrativeML. The NarrativeML can also be used to browse different aspects of the movie, generate narrative variants or segments, vary the narrator characteristics and perspective, and overall, provide fine-grained summarization capabilities. This level of control over automatic narration is unavailable using the generated NL descriptions alone. There are still, however, a number of theoretical and practical challenges.

Keywords

Movie summarization, movie narration, automatic storytelling, narrative representations

1. Introduction

At CMN'2016, a method and annotation scheme was described for inferring the spatial relations and motion events in animated movies [1]. The paper was derived from work mapping spatial relations to natural language [2]. The annotation scheme, called NarrativeML, represented entities and events, spatial and temporal relationships, along with a level of plot and audience reactions. While the scheme had been applied to a variety of animations, the annotations, carried out by hand, proved at the time to be extremely detailed and practically infeasible for more than a few short videos. Nine years later, much has changed, thanks to the advances of Generative AI in analyzing text and images and generating text. In this position paper, we focus not on spatial relations in movies but on narration itself. Effectively auto-narrating movies could of course be useful for the visually impaired and for adding additional dimensions of experience to silent films.

Even though Generative AI can provide narration in both text and speech for existing and newly created videos, the depth of its understanding of film narrative remains unclear, in part due to the opacity of neural nets. We remedy this in part by a system that auto-generates, as an intermediate representation, the NarrativeML and then generates the audio narrative from it. The machine-generated annotation can be used to assess the narrative understanding underlying the generated NL narrative. The intermediate layer of NarrativeML can also be used to browse different aspects of the movie, generate narrative variants or segments, vary narrative distance and perspective, and overall, provide fine-grained summarization capabilities. Such manipulation is clearly impossible using the textual descriptions alone.

This paper discusses this narrative-structure mediated approach. Since automatic video and image generation itself (aside from deep fake copying) is still immature, we restrict the application to cases where the narrative is altered while the images remain the same.

2. NarrativeML

Explored in [1], [4], and [5], NarrativeML is a narrative representation involving multiple layers of annotation, relying on events and their temporal relations represented using TimeML [6], which in

[*] *CMN'25: Workshop on Computational Models of Narrative*, May 28-30, 2025, Geneva, Switzerland.

inderjeet.mani@gmail.com

turn leverages primitives from the temporal interval calculus. NarrativeML also includes a temporal ordering of events for each protagonist, called Narrative Event Chains (NECs) [7]. References to places and static relations between them are modeled using geographical markup from SpatialML [8] and primitives from qualitative spatial calculi [2]. All these concepts form part of the *fabula* (or story). NarrativeML also represents the mapping to *sjuzhet* (or discourse), including the seven varieties of ordering described by Genette [9], as well as narrative tempo and subordinated discourse. In addition, it includes character goals, pre-conditions and post-conditions on events (using open-domain relations rather than any fixed set of primitives), and evaluations by an assumed reader of event outcomes for a particular character.

> *Tina made spaghetti for her boyfriend. It took a lot of work, but she was very proud. Her boyfriend ate the whole plate and said it was good. Tina tried it herself, and realized it was disgusting. She was touched that he pretended it was good to spare her feelings.*

Example 1: Tina's Story

Example 1 shows a simple text story from the ROC story corpus [3]. We give this story as an input prompt to GPT-4o (the Gen AI used throughout this paper), preceded by the DTD and a single-shot training example: the sentence "*March 7, 2006. Leaving San Cristobal de las Casas, I biked with Gregg and Brooks for one more day*", and its hand-curated NarrativeML. In Figure 1, we show the output NarrativeML from the Gen AI[1]:


```xml
<NarrativeML version="2.1">
    <NARRATIVE id="n1" title="Tina's Spaghetti" medium="descriptive_text">
      <NARRATOR id="n2" type="present" coref="c1" exists="true" order="CHRONICLE" distance="DIRECT"
              perspective="INTERNALLY_FOCALIZED" timeRelation="SUBSEQUENT" />
        <CHARACTER id="c1" name="Tina" type="animate" exists="true" accessibleTo="c1" />
        <CHARACTER id="c2" name="Boyfriend" type="animate" exists="true" accessibleTo="c1" />
        <CHARACTER id="c3" name="Spaghetti" type="inanimate" exists="true" accessibleTo="c1" />
        &lt;SEGMENT id="s1" title="Cooking and Eating Spaghetti"&gt;
            <EVENT id="e1" participants="c1 c3" duration="PT2H">made</EVENT>
            <CONDITION id="cond1" event="e1" type="pre" logic="Exists(x) (Ingredient(x) & Has(c1, x))" />
            <CONDITION id="cond2" event="e1" type="post" logic="Cooked(c3)" />
            <EVENT id="e2" participants="c1" duration="PT1M">proud</EVENT>
            <CONDITION id="cond3" event="e2" type="pre" logic="Cooked(c3)" />
            <CONDITION id="cond4" event="e2" type="post" logic="Proud(c1, e1)" />
            <EVENT id="e3" participants="c2 c3" duration="PT15M">ate</EVENT>
            <CONDITION id="cond5" event="e3" type="pre" logic="OnPlate(c3)" />
            <CONDITION id="cond6" event="e3" type="post" logic="not(OnPlates(c3, Plate))" />
            <EVENT id="e4" participants="c2" duration="PT1M">said</EVENT>
            <CONDITION id="cond7" event="e4" type="pre" logic="Ate(c2, c3)" />
            <CONDITION id="cond8" event="e4" type="post" logic="Believe(c1, Good(c3))" />
            <EVENT id="e5" participants="c1" duration="PT2M">tried</EVENT>
            <CONDITION id="cond9" event="e5" type="pre" logic="not(Empty(c3))" />
            <CONDITION id="cond10" event="e5" type="post" logic="Tasted(c1, c3)" />
            <EVENT id="e6" participants="c1" duration="PT1M">realized</EVENT>
            <CONDITION id="cond11" event="e6" type="pre" logic="Tasted(c1, c3)" />
            <CONDITION id="cond12" event="e6" type="post" logic="Aware(c1, not(Good(c3)))" />
            <EVENT id="e7" participants="c1" duration="PT1M">touched</EVENT>
            <CONDITION id="cond13" event="e7" type="pre" logic="Complimented(c2, c3)" />
            <CONDITION id="cond14" event="e7" type="post" logic="Touched(c1, Complimented(c2, c3))" />
            <TLINK id="tr1" type="BEFORE" eventID="e2" relatedToEvent="e3" />
            <TLINK id="tr2" type="BEFORE" eventID="e3" relatedToEvent="e5" />
            <TLINK id="tr3" type="SIMULTANEOUS" eventID="e6" relatedToEvent="e7" />
        &lt;/SEGMENT&gt;
        <PLOT id="p1" NECS="nec1 nec2 nec3 nec4 nec5 nec6 nec7" />
        <NEC id="nec1" entity="c1" events="e1 e2 e5 e6 e7" />
        <NEC id="nec2" entity="c2" events="e3 e4" />
        <NEC id="nec3" entity="c3" events="e1 e3 e5" />
        <EVALUATION id="ev1" eventID="e4" characterID="c2" audienceID="reader1" value="1" polarity="positive" />
        <EVALUATION id="ev2" eventID="e6" characterID="c1" audienceID="reader1" value="1" polarity="negative" />
        <GOAL id="g1" parent="g2" character="c1" leaf="true" events="e1">cook</GOAL>
        <GOAL id="g2" parent="" character="c1" leaf="false">show love</GOAL>
        <GOAL id="g3" parent="g2" character="c1" leaf="true" events="e2">feel proud</GOAL>
        <GOAL id="g4" parent="" character="c2" leaf="false">show appreciation</GOAL>
        <GOAL id="g5" parent="g4" character="c2" leaf="true" events="e4">say</GOAL>
        <TEMPO storyTime="PT2H30M" readingLength="PT45S" />
    </NARRATIVE>
</NarrativeML>
```


Figure 1: Auto-generated NarrativeML fragment for Tina's Story

[1]The latest NarrativeML DTD lives at https://tinyurl.com/5akfxsvs. For reasons of space, spatial relations, tag offsets, and other details are left out of the XML.

It can be seen that the story is internally focalized, namely the narrator sees what Tina knows, while having access to her thoughts. (The narrator isn't actually present in the story, and direct speech is not used but indirect speech is – both of these are errors in the Gen AI output). There are three entities mentioned: Tina, the boyfriend, and the spaghetti. Tina has a goal g2 of showing love, which has a subgoal g1 of cooking (event e1), after which she has a subgoal g3 of feeling proud (e2). There are also events of the boyfriend eating the spaghetti (e3), followed by his saying (e4) it's good, motivated by his goal (g4) of showing appreciation, then her trying it (e5), and realizing (e6) that it's not good, and being touched (e7). Each of these events have pre- and post-conditions, and the events are in chronological sequence except for the simultaneous events of realizing and being touched. Tina participates in a sequence of five events, the boyfriend in two, and the spaghetti in three (indicated by the NECs), with obvious intersections. As for evaluations, the reader appreciates the boyfriend's compliment (e4), and is upset at Tina's realizing (e6) that it doesn't taste good. There are also estimated durations for events and for the story as a whole and its reading[2].

3. Automatic Movie Narration System

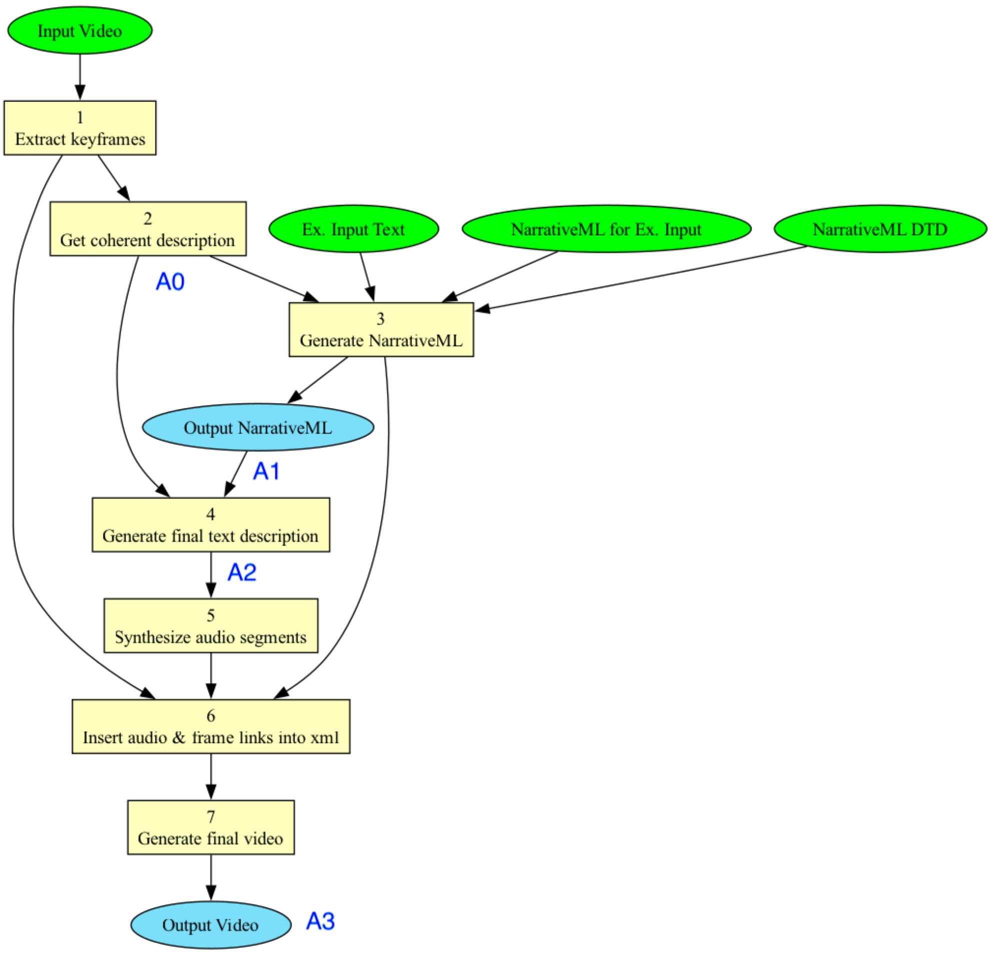

Figure 2: System Architecture in Seven Steps

[2]Further prompting about the plausibility of Tina's Story elicits the following Gen AI judgment invoking taste preferences: *While Tina finds the spaghetti disgusting, her boyfriend might genuinely like it, meaning that her assumption could be wrong. In that case, he might not be pretending at all, and her interpretation of his intentions would be inaccurate.*

As shown in Figure 2, the system takes in the video of a movie. Any audio present is ignored. Here are the processing steps (Gen AI use is underlined): (1) Keyframes are extracted from the video using uniform sampling with a skip-frame heuristic, aiming for *n* candidate frame timestamps by dividing the video duration by *n*, and at each timestamp, grabbing a frame (using *FFmpeg*). If the chosen frame is too similar (based on the perceptual hash Hamming distance between frames) or too close in time to the previous one, the algorithm skips forward until it finds another admissible frame or it runs out of frames. (2) Once the keyframes are extracted, a coherent text description is obtained for each frame (shown as A0), using the Gen AI system in zero-shot mode. (3) The sequence of frame descriptions is passed again to Gen AI for NarrativeML generation. Here we provide as a one-shot example the NarrativeML DTD, a short example input text (Tina's story, from Example 1), and the NarrativeML for it (shown earlier in Figure 1). (4) The new NarrativeML (generated by the Gen AI), shown as A1, is passed as input, along with the sequence of descriptions, yet again to the Gen AI to generate a single-para final text story (A2) for the video as a whole, with the frame indices preserved. (5) That final text description is passed to a speech synthesizer (Google's *gTTS*) to generate audio for each frame. (6) The audio and frame links are added to the NarrativeML, and then (7) the audio sequence is input to the final video generator (using *FFmpeg)*, which delays the next frame until the audio for that frame is completed, producing a narrated video. The NarrativeML (A1) is also output from the system along with the generated video (A3).

The videos we have used are constrained by token limits in the generative AI system to be typically less than three minutes long. For keyframe extraction in this environment, sampling a target dozen frames seems to work quite well, as does a Hamming distance threshold of 5.

4. Example Auto-Narrated Film

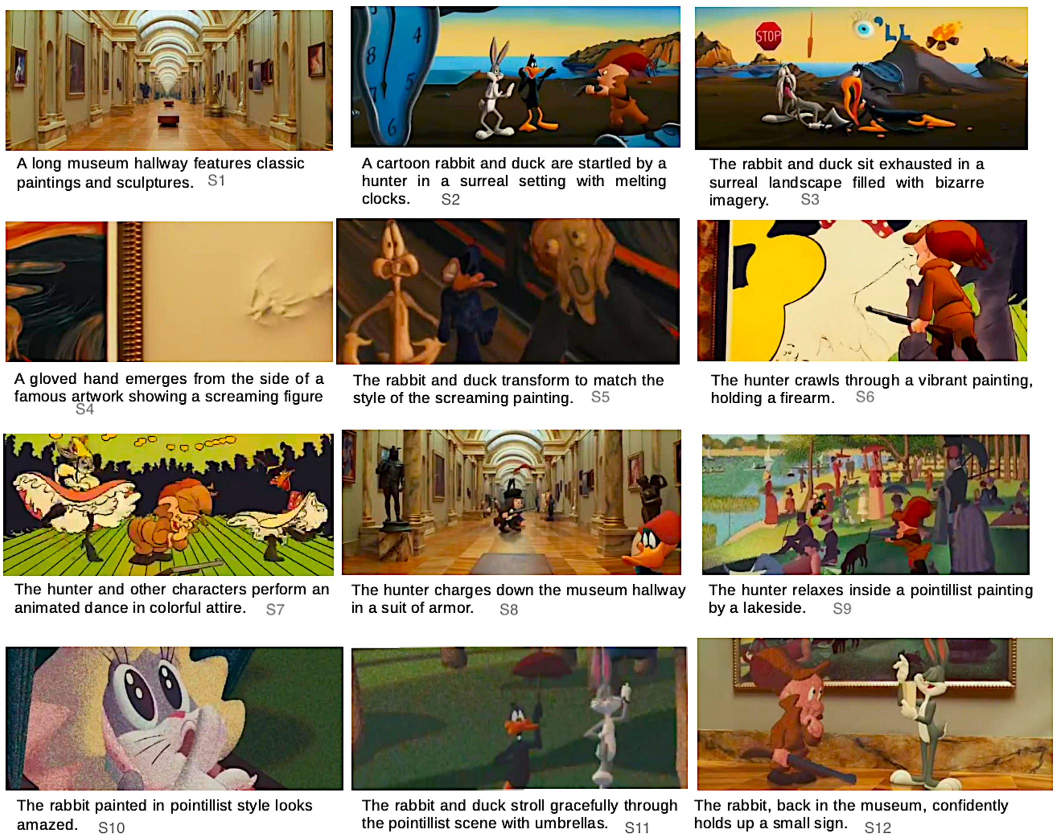

Figure 3: Auto-generated image descriptions from movie

For the choice of movies, we use personal phone videos as well as extracts of YouTube videos including cartoons that lack significant identifying information in the extracts. The reason is that if one uses easily recognizable videos, one cannot adequately test the image description and NarrativeML capabilities as these will likely depend on learned information specific to those videos present in the Gen AI training data or other meta-information found via web search. Since private phone videos can be opaque to the reader, we present an example that may be known to humans but that is still relatively new to the Gen AI system. (This can be confirmed by prompts: if suggestive information about movie titles or other such information is provided, many more specific details are provided in the Gen AI's image descriptions, including names of the characters, the story line, and comments from reviewers.) The example video is a pre-existing 2.07-minute YouTube excerpt from the Bugs Bunny movie *Looney Tunes: Back in Action*[3], which is one of the films discussed in [1]. Figure 3 shows the frame descriptions generated by the system (corresponding to output A0 in Figure 2). It can be seen that the Gen AI has done a creditable job describing the images, a point that should not be surprising to this audience.

```
<CHARACTER id="c1" name="Rabbit" type="animate" form="cartoon"
           attributes="startled, exhausted, amazed clever, proactive"/>
<CHARACTER id="c2" name="Duck" type="animate" form="cartoon"
           attributes="startled, exhausted"/>
<CHARACTER id="c3" name="Hunter" type="animate" form="cartoon"
           attributes="aggressive, puzzled"/>
<GOAL id="g1" parent="" character="cl c2" leaf="false" events="el e3 eS e7">navigate surreal world</GOAL>
<GOAL id="g2" parent="" character="c3" leaf="false" events="e2 e6 e8 el2">capture rabbit and duck</GOAL>
&lt;SEGMENT id="s1" title="Museum Hallway">
&lt;SEGMENT id="s2" title="Startled Encounter">
&lt;SEGMENT id="s3" title="Exhausted in Surreal Landscape">
&lt;SEGMENT id="s4" title="Screaming Artwork">
&lt;SEGMENT id="s5" title="Transformed in Style">
&lt;SEGMENT id="s6" title="Hunter in Vibrant Painting">
&lt;SEGMENT id="s7" title="Animated Dance">
&lt;SEGMENT id="s8" title="Hunter Charges in Armor">
&lt;SEGMENT id="s9" title="Pointillist Painting">
&lt;SEGMENT id="s10" title="Rabbit's Amazement in Pointillist Style">
&lt;SEGMENT id="s11" title="Graceful Walk with Umbrellas">
&lt;SEGMENT id="s12" title="Confident Reveal">
<NEC id="necl" entity="c1" events="el e3 e5 e10 e11"/>
<NEC id="nec2" entity="c2" events="el e3 e5 e11"/>
<NEC id="nec3" entity="c3" events="e2 e6 e8 e9 e12"/>
<EVALUATION id="ev1" eventID="e2" characterID="c1" audiencelD="reader1" value="1"
            polarity="negative">
<EVALUATION id="ev2" eventID="e5" characterID="c1" audiencelD="reader1" value="1"
            polarity="positive">
<EVALUATION id="ev3" eventID="e12" characterID="c1" audiencelD="reader1" value="1"
            polarity="positive">
```

Figure 4: Auto-generated NarrativeML fragment for Bugs Bunny movie

A small fragment of the auto-generated NarrativeML (corresponding to output A1 in Figure 2) for the Bugs Bunny excerpt is shown in Figure 4. The segments (s1, s2, etc.) are in one-to-one correspondence to the scene labels from Figure 3. For reasons of space, the events, temporal relations, and pre- and post-conditions are elided, but there is one event per segment, with the same numbering, viz., e1 corresponds to s1, e2 to s2, etc. The Gen AI produces an error in one of the temporal relations, where a pair of events in two successive scenes is marked as simultaneous rather than preceding. The character goals and the Narrative Event Chains (NECs) for each character, which are of course absent in the text descriptions, are very much present in the NarrativeML. However, the character goals are just leaf nodes, with no higher-level goal structure, which is also an error.

Turning to character evaluations, the Gen AI believes the reader feels negative about the outcome for the rabbit in the early scene 2 (with the startled rabbit), positive about the rabbit in scene

[3] https://www.youtube.com/watch?v=97PLr9FK0sw .

5 (where the two animals are transformed into entities in the Munch painting), and positive about the rabbit in the final scene. No evaluations are provided for the duck or the hunter, which is another error. Is the AI just guessing randomly at character evaluations? Quite the opposite. Further prompting, incurring additional token costs, does generate further (and correct) evaluations for the other characters over additional scenes, though non-determinism in the AI means that not everything from the earlier versions will be preserved exactly. Although it can’t be established even anecdotally in this paper without showing additional examples, research in [5] reveals that the Gen AI does have a partial grasp of character evaluations, which form, as argued in [10] and [11], a key aspect of a reader’s understanding of story.

The final text description for the entire video (corresponding to output A2 in Figure 2), which is eventually turned into audio and synchronized with the frames into the final video, is shown in Table 1. The Gen AI has clearly homed in on the fact that this is a <u>pursuit</u> story, including the correct sequence of events and alluding to the character goals as well as the event outcomes. The text that clearly derives from the NarrativeML layer is underlined: the absent narrator (inferred correctly this time) is one example, and the grouping of character attributes is another, as are the character evaluations. None of these underlined inferences would be possible without NarrativeML.

In this Surreal Cartoon Adventure, <u>an absent narrator</u> introduces a comedic storyline featuring a s<u>tartled yet clever rabbit, a chaotic and clever duck, and a persistent, puzzled, and clumsy hunter.</u> The rabbit and duck strive to evade capture and explore a surreal world, beginning in a museum hallway <u>where their antics soon draw positive audience reactions</u>. As the story unfolds, they undergo a visually creative transformation sequence that further confounds the hunter, reinforcing the comedic tone. The hunter remains determined to capture them, yet his clumsiness and confusion amplify the humor each time his attempts fail. Ultimately, the rabbit and duck execute a confident reveal back in the museum, decisively outwitting the hunter and leaving him foiled.

Table 1: Auto-generated output narrative for Bugs Bunny movie

For the narrated output, we can vary the narrative voice, narrative distance, and focalization by asking the Gen AI to change the output in Table 1, or more interestingly, by directly editing the NarrativeML to add in this information (see the DTD for expressive options).

5. Modifying Movies through NarrativeML

In this version, the rabbit and duck are on a secret mission to retrieve a valuable artifact from the museum. Their presence is detected by the hunter. They pretend to be exhausted, and make use of the museum’s magical properties to transform themselves into part of the artwork. Meanwhile, the hunter stumbles into one of the paintings and becomes stuck. To celebrate their success, the rabbit and duck perform a playful, exaggerated dance within the paintings, which further confuses the hunter. The rabbit and duck emerge from the paintings and stroll confidently through the museum with umbrellas in hand. They hold up a sign that reads, “Better luck next time!” as the hunter, still trapped in the painting, looks on helplessly.

Table 2: Auto-generated new plot for same Bugs Bunny visuals

A given video allows for multiple interpretations and narratives. To encourage storytelling diversity, we can prompt the Gen AI: “*Given this NarrativeML* [from Figure 4] *and this sequence of video* *frames* [the ones in Figure 3], *generate an entirely different story*." The Gen AI output is shown in Table 2. The narrative has been changed from a pursuit to a heist.

While it will be presumptuous to assume that the Gen AI understands its new reimagining of the story, its reasoning nevertheless displays consistency with the new logic. When we give the AI as a prompt the old NarrativeML (of which Figure 4 is a fragment) along with the new plot in Table 2, it returns with a set of changes to NarrativeML. For reasons of space we cannot show the detailed output, but here is a summary: All references to "Surreal Cartoon Adventure" are changed to "The Museum Escape Plot". The goals of the rabbit and duck are changed from "navigate surreal world" to "retrieve valuable artifact," and the hunter's goal from "capture rabbit and duck" to "catch the intruders". Segments like "Startled Encounter" (scene 2), "Exhausted in Surreal Landscape" (scene 3) and "Hunter in Vibrant Painting" (scene 6) are updated to "Feign Startled Encounter", "Pretend Exhaustion in Surreal Landscape" and "Hunter Stumbles and Get Stuck in Artwork", respectively; these updates convey the duo's manipulative ploy. Events and their pre-and post-conditions are reworked to reflect these deceptions. (Thus the Animated Dance – scene 7 – is re-contextualized as one of celebration by the plotters). The character evaluation for scene 2, originally marked as negative for the startled rabbit, is updated to positive, reflecting the duo's change in status from being startled victims to them actively tricking the hunter.

The timeline is also restructured to correct the previous temporal relation error, while also removing certain events (corresponding to scenes 5 and 8) because the AI believes them to be comedic distractions from the new heist plot. This presents two problems: first, the AI can make errors in the new temporal relations. Second, having the AI revise the NarrativeML without having access to the frames can clash with the need to preserve the frame sequence; so instead, both the old NarrativeML and the frame sequence constraint should both be provided,

The resulting movie description the AI generates directly from the revised NarrativeML (corresponding to output A2 in Figure 2) is shown in Table 3, producing in addition a new movie with a different audio narrative that accompanies the same synthesized output frame sequence (corresponding to output A3 in Figure 2). This shows again the value added by the underlying narrative representation.

A clever rabbit and a resourceful duck sneak through a museum to retrieve a valuable artifact, all while a bumbling hunter tries to capture them. Feigning surprise, they lead him deeper into a surreal gallery, where the artwork comes to life. The pair pretends to collapse from exhaustion, allowing the hunter to stumble right into a painting and become stuck. Seizing the moment, the rabbit and duck merge into the artwork themselves, performing a triumphant dance and confusing the baffled hunter even more. Ultimately, they stroll confidently with umbrellas through a pointillist scene and reappear in the museum to display a parting message—leaving the trapped hunter behind as they enjoy their well-earned victory.

Table 3: New Bugs Bunny movie narrative auto-generated from auto-revised NarrativeML

6. Conclusions

These short movie narrations can be embellished with title cards, background music, and sound effects. However, effective use of such bells and whistles requires prompting the Gen AI with specific aesthetic goals and user models. Introducing dialogue into the narration is more challenging, and is left for future work, which can also involve having the AI fuse information from the input audio where available. Image processing remains weak in some areas, and emotion recognition, essential for narrative, is hardly there. Finally, NarrativeML itself needs to be extended further to accommodate film-specific media. Tailoring to different classes of users, such as children and the visually impaired, is an obvious application of this approach.

We have already pointed to various errors in the Gen AI output. Before unleashing such applications on users, new evaluation frameworks are needed for Gen AI's automatic narrative analyses and generated stories. While the inventiveness of the AI is remarkable, strongly suggesting a level of emergent storytelling knowledge, formally evaluating such capabilities requires rich narrative-specific reasoning tasks [5]. In evaluations, comparing narrative structures at different abstraction levels may be preferable to comparing non-unique narrative texts or captions (let alone generated images), since the compared elements may be dissimilar yet valid. However, in many such dense annotation scenarios (for example, TimeBank-DENSE [12]) the necessary inference-based comparisons result in poor agreement. Synthetic datasets may have a role to play here. Finally, another well-known issue is length. Long videos are too expensive to process, and LLMs become incoherent when generating longer stories, easily trapping themselves in narrative dead-ends, much like the mis-spun tales of ancient AI storytellers [13].

Declaration on Generative AI

The author used the GPT-4o API to generate the text shown in Figures 3-4 and Tables 1-3.

References

[1] Mani, Inderjeet. 2016. "Animation Motion in NarrativeML." 7th Workshop on Computational Models of Narrative (CMN 2016), vol. 53, pp. 3:1-3:16. https://doi.org/10.4230/OASIcs.CMN.2016.3

[2] Mani, Inderjeet and James Pustejovsky. 2012. Interpreting Motion: Grounded Representations for Spatial Language. New York: Oxford University Press.

[3] Mostafazadeh, Nasrin, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. "A corpus and evaluation framework for deeper understanding of commonsense stories." Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT'2016), pp. 839-849.

[4] Mani, Inderjeet. 2013. Computational Modeling of Narrative. Synthesis Lectures on Language Technologies. San Rafael, CA: Morgan & Claypool.

[5] Mani, Inderjeet. 2025. Generative AI and Narrative: A Computational Account. Berlin: Springer Nature.

[6] Pustejovsky, James, Bob Ingria, Roser Saurı, Jose Castano, Jessica Littman, Rob Gaizauskas, Andrea Setzer, Graham Katz and Inderjeet Mani. 2005. "The specification language TimeML." In Inderjeet Mani and James Pustejovsky (eds.), The Language of Time: A Reader, pp. 549–562. New York: Oxford University Press.

[7] Chambers, Nathanael. 2011. Inducing Event Schemas and their Participants from Unlabeled Text. Ph.D. Dissertation, Department of Computer Science, Stanford University.

[8] Mani, Inderjeet, Christine Doran, David Harris, Justin Hitzeman, Robert Quimby, Justin Richer, Ben Wellner, Scott Mardis, and Seamus Clancy. 2010. "SpatialML: annotation scheme, resources, and evaluation." Language Resources and Evaluation, vol. 44, no. 3, pp. 263–280.

[9] Genette, Gerard. 1980. Narrative Discourse (trans. Jane Lewin). Ithaca: Cornell University Press.

[10] Herman, David. 2009. Basic elements of narrative. Hoboken, NJ: John Wiley & Sons.

[11] Mani, Inderjeet. 2010. The Imagined Moment. Lincoln, NE: University of Nebraska Press.

[12] Cassidy, Taylor, Bill McDowell, Nathanael Chambers, and Steven Bethard. 2014. An Annotation Framework for Dense Event Ordering. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics}, pp.~501–506. https://aclanthology.org/P14-2082.pdf .

[13] Meehan, James R. 1976. The Metanovel: Writing Stories by Computer. PhD Dissertation, Dept. of Computer Science, Yale University.