# EPICQA: A Counterfactual Evaluation Framework for Story Illustration

Anonymous ACL submission


## Abstract

Image generation models are useful tools for story illustration tasks. However, existing metrics are not well-equipped for evaluating complex aspects of narrative fidelity because they do not capture fine-grained details of the story. Similarly, these metrics cannot effectively probe model sensitivity to different inputs. We introduce EPICQA, an evaluation framework for story illustration models, to address these limitations. EPICQA is built around counterfactual probing: each story is paired with counterfactual versions, altering a specific attribute while leaving the rest of the story intact. Models are scored along two complementary dimensions: *counterfactual sensitivity* (do the output illustrations reflect the counterfactual changes?) and *narrative preservation* (does it leave everything else unchanged?). We instantiate EPICQA on stories from two datasets, Tell-Me-A-Story and Project Gutenberg, and benchmark performance of five state-of-the-art image generation models. In our experiments, we investigate how scores vary across multiple dimensions (e.g. sequence length, and attribute type) which illuminates multiple areas where models need improvement for more robust depictions.


## 1 Introduction

Recent advances in multimodal generative models have made automatic story illustration an increasingly viable creative application: given a text-based narrative, a model can now produce a sequence of images depicting the story's key events (Liu et al., 2024; Yang et al., 2025; Ye et al., 2024; Rahman et al., 2023). Yet evaluating the quality of such illustrations remains a fundamentally harder problem than evaluating single image generation. A story illustration system must faithfully represent the fine-grained attributes of characters, settings, and mood, and maintain visual consistency across the sequence of images. This is a challenge for existing image evaluation metrics which were typically designed to match holistic similarity between text and images rather than picking up on how fine-grained details are being depicted (e.g. Figure 1). Moreover, existing metrics are unable to detect whether a model is genuinely sensitive to the input or relying on internal biases (e.g. relying on stereotypical or memorized characterizations rather than fully conditioning to the input).

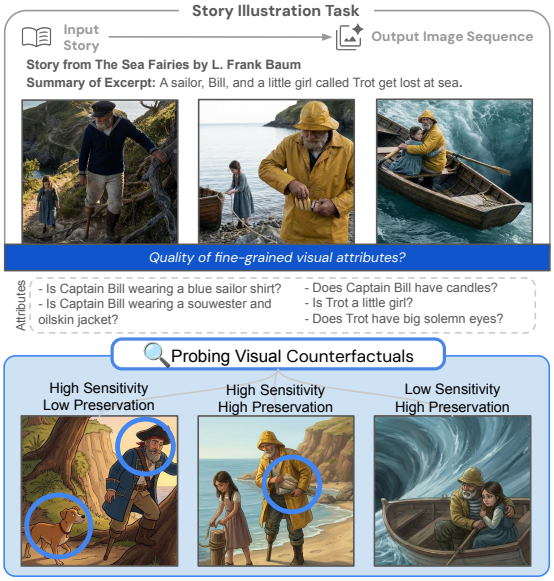


Figure 1: Example of story illustrations and counterfactual images with different levels of sensitivity and preservation of attributes.

In this paper, we introduce EPICQA, a novel framework for evaluating story illustration models to address these limitations by probing counterfactuals as in Figure 1. EPICQA is built around a simple but powerful idea: by pairing each story with *counterfactual* versions that each alter a different character attribute, we can measure two important properties: *counterfactual sensitivity*, whether the model's depictions correctly reflect the altered attributes, and *narrative preservation*, whether the depictions of all unaltered aspects of the story are

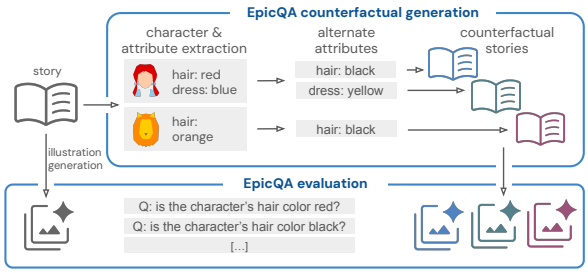


Figure 2: The EPICQA framework consists of two separate steps: the counterfactual data generation based on specific visual attributes, followed by the VQA task checking these attributes.

unchanged. A model that performs well on both is conditioning well to the input text; a model that fails on counterfactual sensitivity while succeeding on narrative preservation is failing to adapt to the input and may instead be relying on internal biases.

The EPICQA framework has two components (Figure 2): an automated pipeline that creates counterfactual (CF) stories by extracting character attributes, generating alternate values, and rewriting stories; and a visual question answering (VQA) based evaluation protocol that uses questions about attributes, similar to those in Figure 1, to directly score both counterfactual sensitivity and narrative preservation for individual attributes. We instantiate EPICQA on stories from publicly available datasets: Tell-Me-A-Story (TMAS; Huot et al. 2025) and Project Gutenberg 2019 (PG; Rae et al. 2019). The resulting benchmark comprises 100 seed stories paired with about 1200 counterfactual stories and a total of 2,500 fine-grained VQA pairs across 250 visual attribute types (e.g., appearance, clothing, age).

We use EPICQA to benchmark five state-of-the-art image generation models: Gemini 2.5 Flash Image (Google DeepMind, 2025a) and Gemini 3 Pro Image (Google DeepMind, 2025b), GPT Image 1.5 (OpenAI, 2025), Qwen Image (Wu et al., 2025), and Flux 2 (Black Forest Labs, 2025). We find that performance on the EPICQA benchmark varies between models, but that there is substantial room for improvement and that performance tends to degrade when producing longer image sequences, demonstrating that this is a challenging task for current SOTA models of varying types. We observe that EPICQA also highlights differences between types of visual attributes, indicating that some attributes (particularly more abstract features) might be more challenging for the illustration models. Lastly, we also validate our framework with human evaluations and compare EPICQA against existing automatic metrics (e.g. Wiles et al. (2025) and an LLM-based auto-rater).

Our main contributions are: (1) the EPICQA framework, a domain-agnostic, automated method for evaluating story illustration models with counterfactual attribute probing; (2) the EPICQA benchmark, a concrete instantiation using stories from TMAS and PG; (3) a systematic evaluation of five state-of-the-art image generation models with comparison to existing automatic metrics, demonstrating that EPICQA provides finer-grained and more diagnostic signal than existing approaches.

## 2 Background

**Counterfactual Stories** Counterfactual reasoning is commonly used in NLP tasks such as generating counterfactual story rewrites (Qin et al., 2019; Hao et al., 2021; Mu and Li, 2024a,b). Counterfactuals are frequently used to create specialized data for testing reasoning capabilities (Ahuja et al., 2025; Wu et al., 2023; Ghosh et al., 2023) or probing sensitivity to input biases (Wu et al., 2024; Ye et al., 2021; Dycke and Gurevych, 2026). We share the same underlying intuition of using minimally-edited pairs to surface model behavior. EPICQA applies counterfactual probing to the multimodal narrative setting, measuring whether an illustration model's visual output is genuinely conditioned on attributes from the input and sensitive to fine-grained changes. Building on prior work, we adapt prompts from the counterfactual story creation pipeline in Ahuja et al. (2025) in our own pipeline.

**Story Illustration Benchmarks** There is a lot of recent progress in developing visually-grounded narrative benchmarks (Gao et al., 2025; Zhuang et al., 2026; Yang et al., 2025; Liu et al., 2024; Ye et al., 2024). Many existing benchmarks either use images to elicit narrative text, often written post-hoc (VIST (Huang et al., 2016), VWP (Hong et al., 2023)) or build from literal scene descriptions using cartoon or video frames with fixed casts of characters (Li et al., 2019; Gupta et al., 2018; Maharana and Bansal, 2021; Rahman et al., 2023). In contrast, EPICQA builds on narratives from existing literary corpora, yielding longer, narratively richer inputs without the literal scene-description bias of post-hoc generated captions. We also evaluate using

counterfactual story pairs rather than single stories, which exposes failure types that aren't covered by existing benchmarks.

Many evaluations for this task rely on metrics (Hessel et al., 2021; Wiles et al., 2025) that were originally designed for matching images to literal text captions. This setting differs from story illustrations where text input is longer and includes superfluous information, that doesn't align to the images. Some character consistency evaluations (Tewel et al., 2024; Zhou et al., 2024) focus more on preserving visual identity across frames rather than measuring alignment between individual attributes and the text. Gao et al. (2025) uses character-attribute alignment as part of their evaluation, similar to our work; however, their task set-up is focused on judging a broader set of commonsense and discourse constraints whereas we focus solely on character attributes and also probe sensitivity to counterfactuals. Many existing metrics are also holistic scores which are not inspectable. Our work is novel in providing metrics that specifically measure the counterfactual sensitivity at a fine-grained attribute level.

The NewEpisode benchmark (Wang et al., 2024), constructs counterfactual visual stories by swapping in *new characters*. While conceptually similar, NewEpisode differs from EPICQA in regards to: (a) the underlying stories are short, narratively simple cartoon scene descriptions with a fixed closed-world cast, while we consider open-domain narratives; (b) its counterfactuals are restricted to character substitution, whereas EpicQA varies a wide set of attributes (e.g., clothes, age, location); and (c) its evaluation centers on using holistic similarity metrics (e.g. FID, CLIP), whereas EpicQA introduces a means of measuring attribute-level diagnostics, accounting for both counterfactual sensitivity and narrative preservation.

## 3 Counterfactual Story Illustration

Automatic story illustration takes a text story $x$ as input and produces a sequence of $n$ images $y = [y^1, \ldots, y^n]$ depicting an ordered sequence of events from the story (see Figure 1). We use counterfactual edits to probe how faithfully model's illustrations reflect the input text.

**Counterfactual Definitions** We take a seed set of text stories and rewrite the text to create a set of counterfactual stories, each of which has one changed attribute. To define the set-up more concretely, for each story, $x$, we first identify a set of visually grounded premises $Z = \{z_1, \ldots, z_k\}$, where each premise $z \in Z$ describes a specific character attribute (e.g., *Bill possesses candles*). Each attribute has a type (e.g., *possession*) and value (e.g., *candles*) and is attached to a story character who it describes (e.g. Bill). We focus on this type of character-based attribute because they affect fine-grained visuals without changing much of the overall plot. For each premise $z \in Z$, we define a counterfactual value, $\hat{z}$, as a plausible alternative to $z$ (e.g., *Bill possesses old maps*). We then construct a counterfactual story $x_{\hat{z}} = \text{cf}(x, z, \hat{z})$, where $\text{cf}(\cdot)$ denotes a rewriting operation that minimally edits $x$ to incorporate $\hat{z}$ in place of $z$, preserving all other premises in $Z \setminus \{z\}$. For instance, in the story from Figure 1, changing Bill's possessions from candles to maps may alter the text by substituting his action on shore from lighting candles to reading maps, but shouldn't change any other aspects of the story. For each story $x$, this yields a set of counterfactual stories $\{x_{\hat{z}_1}, \ldots, x_{\hat{z}_k}\}$, one per altered premise.

**Task Objective** Given a counterfactual story $x_{\hat{z}}$, the task consists of generating an illustration sequence $y_{\hat{z}} = [y^1_{\hat{z}}, \ldots, y^n_{\hat{z}}]$ that satisfies two conditions simultaneously:

1. **Counterfactual sensitivity:** $y_{\hat{z}}$ correctly reflects the altered premise $\hat{z}$ (i.e., it differs from $y$ in the visual dimensions corresponding to $\hat{z}$).
2. **Narrative preservation:** $y_{\hat{z}}$ remains faithful to all unaltered premises $Z \setminus \{z\}$ (i.e., it agrees with $y$ on all visual dimensions not targeted by the counterfactual).

These two conditions are in tension for models that generalize poorly: such models may reproduce $y$ faithfully on original stories while failing to adapt to $x_{\hat{z}}$, satisfying narrative preservation but violating counterfactual sensitivity. EPICQA (Section 5) is designed to measure both conditions jointly.

## 4 The EPICQA counterfactual pipeline

We create the EPICQA dataset via an automated pipeline, using Gemini 2.5 Flash as the backbone LLM at each stage (validating quality in Sec 4.2; prompts in App. C). The pipeline takes an original story $x$ as input and produces a set of counterfactual stories $\{x_{\hat{z}_1}, \ldots, x_{\hat{z}_k}\}$, one for each identified character attribute. It is designed to be domain-agnostic for stories of varying length and style.

**Stage 1: Character and Attribute Extraction** We use a zero-shot prompt to extract the names of all characters appearing in $x$. The prompt also instructs the model to extract a list of their attributes, defined as any identifying character trait that is described in the text.

**Stage 2: Alternate Attribute Generation** For each extracted attribute $z$, we use a zero-shot prompt to generate an alternate value $\hat{z}$, conditioned on the story $x$. The model is instructed to produce a value that is (a) meaningfully different from $z$, (b) plausible with the rest of the story, and (c) of the same type as $z$ (e.g., a color attribute should be replaced with a different color).

**Stage 3: Counterfactual Rewriting** For each attribute-alternate pair $(z, \hat{z})$, we rewrite the story $x$ into a counterfactual story $x_{\hat{z}}$ using the counterfactual story rewriting prompt from Ahuja et al. (2025), which prompts the model to substitute $\hat{z}$ for $z$ throughout the narrative while making only minimal edits and preserving coherence. Crucially, all other premises $Z \setminus \{z\}$ are unchanged, ensuring that $x_{\hat{z}}$ differs from $x$ only in the targeted attribute.

**Stage 4: Attribute Scoring** We score each attribute on two dimensions using zero-shot prompts: its *plot relevance* (PR; a 5-point scale where 1 means the attribute is incidental to the plot and 5 means it is central to the plot) and its *visual nature* (VN; a 5-point scale where 1 means the attribute has minimal impact on how a character should be visualized, and 5 means it is integral to their visual features). These scores are used in Section 7 for fine-grained analysis. In the evaluation framework, we filter to only use attributes with VN $> 2$ to limit to visually-oriented counterfactuals.

### 4.1 Seed Data and Dataset Statistics

We seed the pipeline with stories from two publicly available datasets chosen to contrast along two axes: story length and likelihood of memorization. **Tell-Me-A-Story** (TMAS; Huot et al. 2025) consists of shorter stories that were created by recent work. **Project Gutenberg** (PG;[1] `www.gutenberg.org`) consists of longer, canonical literary texts. We randomly select 50 stories from each source.

Table 1 summarizes the key statistics of the two sources. TMAS stories are shorter on average than PG stories, with fewer characters per story and fewer attributes per character. Despite these differences, the average visual nature (VN) and plot relevance (PR) scores are nearly identical, suggesting that differences in downstream EPICQA performance across the two sources are more likely attributable to other differences in the domain (e.g. story length and memorization issues) rather than to differences in attribute type. As shown in Table 1, applying the pipeline to all 100 seed stories yields 2.4k counterfactual stories in total; after filtering to VN $> 2$ (to exclude traits that are difficult to assess visually), this reduces to about 1.2k counterfactual stories. The dataset provides a test bed spanning multiple visual attribute types, story lengths, and domains.

| | TMAS | PG |
|---|---|---|
| Story Length (tokens) | $1624 \pm 563$ | $3038 \pm 1428$ |
| # Characters (per story) | 3.84 | 5.48 |
| # CFs (per story) | 29.34 | 48.24 |
| VN Score | 2.46 | 2.47 |
| PR Score | 3.30 | 3.22 |
| # original stories | 50 | 50 |
| # CF stories (VN$>$ 2) | 470 | 769 |
| # QA | 940 | 1538 |

Table 1: EPICQA dataset statistics averaged per story for Tell-Me-A-Story (TMAS) and Project Gutenberg (PG): Story length in tokens (mean $\pm$ std); # Characters and # CFs are per-story means; VN and PR scores are mean visual nature and plot relevance of extracted attributes (1–5 scale).

### 4.2 Quality Control

To validate the pipeline, we conducted a manual audit of 30 randomly sampled counterfactual stories, checking (a) the counterfactual story makes sense on its own, (b) the counterfactual attribute was fully incorporated and (c) the counterfactual story only makes minimal edits, without changing other information. Examples were rated by five NLP experts, who each rated five stories in common with each other and five unique stories independently. On the stories rated in common, pairs of experts agreed in 92%/88%/92% of cases for questions (a)/(b)/(c) respectively, showing very few disagreements, overall. Overall, (A) 93.3% of the stories were judged to make sense fully. (B) 83.3% of stories fully incorporated counterfactual information and the remaining 16.7% all at least partially incorporated the counterfactual information. We find that the most common failure mode was when counterfactual information is only changed

[1] We restrict to stories tagged as English Fiction and use only the first chapter of each, truncated to 5000 words, using the PG-19 version (Rae et al., 2019).

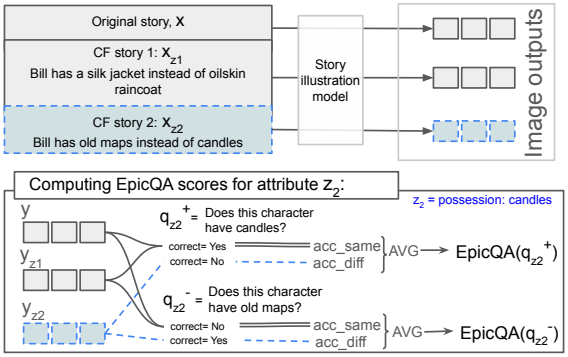


Figure 3: Diagram of how to compute EpicQA score for one target attribute. The accuracy is measured separately for the images from the target counterfactual story vs the original + all other counterfactual stories. This is used to create two scores that are averaged together.

where the attribute is directly mentioned without changing other information that is implicitly related (e.g. when changing a character's house to a large mansion, then parts of the story about their poor financial situation are implicitly related). (C) 90.0% of the rewrites are minimal changes without editing extra content. Based on the expert evaluations, we deem the counterfactual story texts to be of sufficient quality.

## 5 The EPICQA Evaluation Framework

The EPICQA evaluation metric scores a story illustration model's robustness to illustrate both the original story and its counterfactual variants by comparing the attribute values' depictions across both the original and counterfactual illustrations. First, the target illustration model that we are evaluating is used to generate images for each story variant in the EPICQA dataset. Then, we score QA (question/answer) pairs based on how well each attribute is represented across the images in the original vs. counterfactual stories.

**Attribute Questions** Inspired by QA-based metrics (Wiles et al., 2025; Eyal et al., 2019), we create QA-style rubrics for checking attributes in the images. For each attribute in the EPICQA dataset with original value $z$ and counterfactual value $\hat{z}$, we create a pair of questions[2]; one asking whether $z$ is true ($q^+$) and one asking whether alternative premise $\hat{z}$ is true ($q^-$). For the running example from Figure 1, two generated questions would be "Does the character have a possession that is candles?" and "Does the character have a possession that is old maps?". In practice, these are each treated as three-choice questions: Yes, No, or N/A (if unanswerable from the image). The reason we create two separate questions (one from $z$ and one from $\hat{z}$) is to create a balanced QA set with a mix of yes/no answers. We apply this procedure to all character-attribute pairs extracted during data creation, restricting to attributes with visual nature score VN $> 2$. This yields a total of 2,478 QA pairs for both datasets (see bottom row in Table 1).

As shown in Figure 3, when grounding to images generated from the original story and all the other counterfactual variants (e.g. $y_{z_1}$), a QA model should be expected to answer the positive question, ($q^+_{z_2}$), with "yes", reflecting that the attribute value in those stories is unchanged. Conversely, the answer to the same question should be "no" when grounding on the illustrations from the target counterfactual story, reflecting the changes made. Following this intuition, when computing the EPICQA score for each question about attribute $z$, we average two accuracy terms separately: one computed over the VQA accuracy on the illustrations from the original and $Y \setminus \{y_z\}$ stories (in the equations, we term this $\text{Acc}_{\text{same}}$), and one computed on the accuracy of the flipped value on the target counterfactual story ($\text{Acc}_{\text{diff}}$). This average represents the tradeoff between preservation and sensitivity towards this particular attribute value across different variants of the story. By computing separate EPICQA scores per attribute value, we can inspect the fine-grained performance based on different attribute types, allowing for in-depth analysis (Sec. 7).

**Automated Evaluation** More formally, for each question $q_z$ from a given story, we aggregate the scores over all the story variants by computing:

$$\text{EPICQA}(q_z) = \frac{1}{2}\left(\text{Acc}_{\text{same}}(q_z) + \text{Acc}_{\text{diff}}(q_z)\right) \quad (1)$$

where $\text{Acc}_{\text{same}}$ is the accuracy of the model in predicting the original answer to the question when grounding on illustrations from stories where the given attribute is unchanged (i.e. the original story + all counterfactual stories in $Z \setminus \{z\}$) and $\text{Acc}_{\text{diff}}$ is the accuracy of the flipped answer on the illustrations from the counterfactual story created with the target attribute. Here, we define:

$$\text{Acc}_{\text{same}}(q_z) = \frac{1}{|Y_s|}\sum_{y \in Y_s} \mathbb{1}[\text{VQA}(y, q_z) = c] \quad (2)$$

[2]using templates of the style "Does this character have `attribute-type` that is `attribute-value`?"

Where $Y_s = Y \setminus \{y_z\}$, i.e. all the images from stories where the attribute is unchanged and $c$ is the correct answer (yes or no) according to the original text, and VQA refers to the automatic answer prediction from a VQA (visual question answering) model. And:

$$\text{Acc}_{\text{diff}}(q_z) = \mathbb{1}[\text{VQA}(y_z, q_z) = \neg c] \quad (3)$$

The $\text{Acc}_{\text{same}}$ term measures *narrative preservation*: whether the illustration model faithfully depicted visually grounded details that were not changed by the counterfactual edit. $\text{Acc}_{\text{diff}}$ measures *counterfactual sensitivity*: whether the model updated the relevant visual attribute in response to the altered story premise. The overall EPICQA score is the averaged EPICQA$(q_z)$ score over all questions from all attributes in the EPICQA dataset.

For each question-story pair, we evaluate the attribute depiction by automatically answering the question on each individual image in the story sequence and averaging over the entire sequence, excluding any questions that were answered as N/A. To handle multi-character scenes, a preprocessing step automatically draws a bounding box around the target character in each image before giving it to the VQA model, disambiguating the target from other characters in the scene (refer to App. D). For automatically answering the question, we use a multimodal LLM (Gemini 2.5 Flash) as a VQA system.

## 6 Experiments

We use EPICQA to evaluate five off-the-shelf story illustration models: Gemini 2.5 Flash Image (Google DeepMind, 2025a), Gemini 3 Pro Image (Google DeepMind, 2025b), GPT Image 1.5 (OpenAI, 2025), Qwen Image (Wu et al., 2025), and Flux 2 (Black Forest Labs, 2025). We generate illustration sequences of length $n \in \{1, 3, 5\}$. We also compare model outputs against a naive baseline with blank images and an oracle setting in which EPICQA is computed from the story text rather than generated images.

### 6.1 Validating EPICQA Scoring

Because EPICQA relies on a model to automatically answer the QA as part of its scoring, we validate its VQA judgments against human annotations. We describe the human user study instructions in App. I.2.

| | Human IAA | | VQA Accuracy | |
|---|---|---|---|---|
| **Subset** | **PA** | **K$\alpha$** | **Gemini** | **Random** |
| All options | 63% | .43 | 65% | 39% |
| Yes/No only | 85% | .67 | 79% | 53% |

Table 2: Human inter-annotator agreement reported as pairwise percent agreement (PA) and Krippendorff's $\alpha$ (K$\alpha$), and VQA model accuracy graded by human majority vote, for all response options and for the yes/no subset. The random baseline samples an option proportional to the human response distribution (100 trials).

We annotated 160 image-question pairs using the illustrations that were generated by Gemini 2.5 Flash Image. We selected 100 randomly sampled image-question pairs from each of TMAS and PG, stratified across original and counterfactual stories. We filtered only the 160 cases where the VQA model returned a yes/no response since those are what contribute to the EPICQA score. Raters saw the same set-up as the VQA model: a single generated image with a bounding box around the character and one EPICQA question. Annotators were instructed to answer the question using options: *yes*, *no*, *I can't tell from the image*, or *unclear bounding box*. We collect three ratings per image-question pair and compare VQA response against the human majority vote response to compute the accuracy of the QA model.

Human agreement is moderate over all response options (63%; Krippendorff's $\alpha = 0.43$), and higher when restricted to yes/no responses (85%; $\alpha = 0.67$), the cases most relevant to EPICQA. In the right half of Table 2, we report VQA model accuracy (scored by the human majority vote to the same question). The Gemini-based model achieves 65% accuracy overall (including cases that humans judged to be unanswerable). However, focusing on just the subset where humans gave a clear yes or no response, accuracy rises to $\sim 80\%$.

These results show that the VQA model is a reliable, if imperfect, proxy for human judgment on the binary questions that underpin EPICQA.

### 6.2 Benchmarking EPICQA performance

We report the EPICQA scores for the full set of model outputs in Figure 4 as a function of illustration sequence length $n$. First, we observe that EPICQA performance seems fairly dependent on model family, with the proprietary models scoring higher than open-source models across most settings. We also see that output sequence length (n)

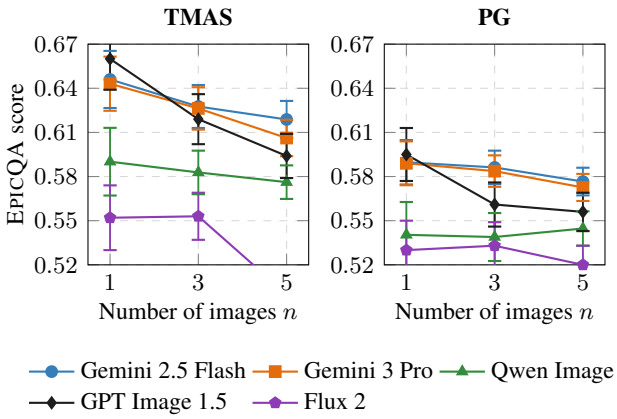


Figure 4: EPICQA score (averaged per question across the dataset) for five story illustration models on TMAS (left) and PG (right) as a function of illustration sequence length $n \in \{1, 3, 5\}$. Error bars show 95% confidence intervals.

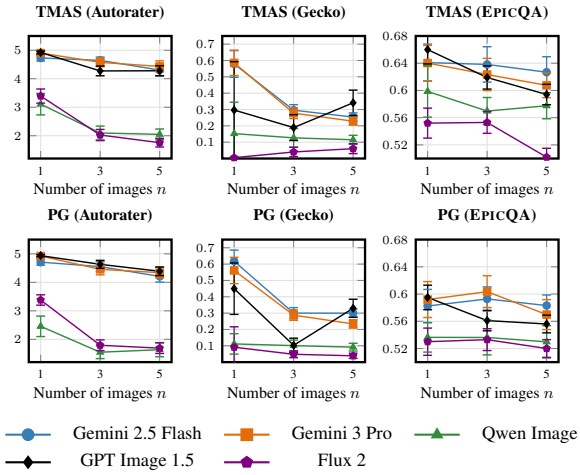


Figure 5: Automatic evaluation metrics as a function of illustration sequence length $n \in \{1, 3, 5\}$. Row 1: TMAS; Row 2: PG. Columns: LLM autorater, Gecko, and mean EPICQA score (averaged across stories). Error bars show 95% confidence intervals.

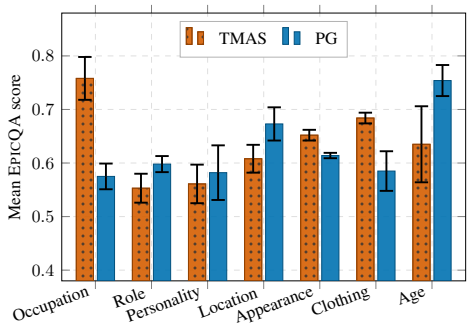


Figure 6: Mean EPICQA consistency score per attribute type for **Gemini 2.5 Flash**, split by dataset (TMAS vs. PG), aggregated across all sequence lengths.

affects the EPICQA score differently depending on the model, but that there is generally a downward trend where models score lower when generating longer sequences. All five models outperform the naive baseline, which is around 49.9/50.0% for PG/TMAS sets respectively. The upper-bound performance (the oracle version of EPICQA) is about 89.6/89.7% performance, demonstrating that there is ample opportunity for these models to improve on this benchmark. Lastly, we note that all models' performance tend to be marginally lower on PG than TMAS portions of the data. All five models also perform more similarly to each other on the PG data than on the TMAS data. This may reflect increased difficulty on Project Gutenberg stories, which are longer and typically of a harder reading level. It also may reflect certain biases that make models less adaptive to counterfactuals on the Project Gutenberg stories (such as memorization biases). Anecdotally, we see that models can be insensitive to counterfactual changes to well-known characters that appear in the PG data (e.g. changing Sherlock Holmes hat or facial appearance).

### 6.3 Comparing to existing automatic metrics

We compare EPICQA against two existing metrics: an LLM-based autorater and Gecko. For apples-to-apples comparison, we average the EPICQA scores per story to get a single score per story, as in other metrics. Gecko (Wiles et al., 2025) is a VQA-based metric that generates binary questions about the input text and checks whether each is answered correctly in the generated image. We also use an LLM-based autorater (Gemini; full prompt in Appendix E) which scores the illustration sequence along five dimensions – stylistic cohesion, narrative fidelity, composition, technical quality, and prompt adherence – each on a 1–5 scale, and averaging to get an overall score.

Figure 5 reports results for all models on both TMAS and PG across the automatic metrics. The LLM autorater reveals a large gap between open-source and proprietary models but doesn't show differences between models beyond that. Moreover, the proprietary models all score above 4.2 out of 5 on the autorater scoring system, not leaving much room for improvement. On Gecko, the scores seem to be overwhelmingly $< 0.5$, which is worse than random chance in the Gecko evaluation framework. This may be due to the domain difference; Gecko was originally intended for comparing images to short, literal captions, and the Gecko QA

generation and filtering process may not adapt well to longer, more abstract narrative text inputs (examples with discussion in App. G). Because EPICQA generates questions from structured character attributes and uses filtering based on visual (VN) scores, we find that it's more robust to the narrative format than Gecko; it filters out fewer questions per story and is more likely to use questions that capture complex visual relationships.

EPICQA is the only of these metrics that is actively probing the differences in portrayals across different counterfactuals or using character-attribute information. This allows for EPICQA to capture unique information that isn't found in other metrics, which we use for quantitative analysis in the following section.

## 7 Fine-grained analysis with EPICQA

One of the key advantages of EPICQA over holistic metrics is its ability to decompose performance along meaningful dimensions, enabling a more granular understanding of where illustration models succeed and where they fall short. In this section, we exploit this property to investigate how attribute type affects EPICQA performance. We also investigate the effect of story length and reading level in the appendices (App. F).

**Attribute types.** We select seven of the common attribute types, and group together EPICQA scores. Figure 6 shows EPICQA broken down by attribute type, split by dataset. Concrete visual features such as "clothing" and "age" score typically higher, while more abstract attributes like "role" and "personality" score lower. However, there are also some nuanced differences across the two data sources. "Occupation" shows the largest dataset gap, (0.76 on TMAS; 0.58 on PG). "Age" shows the reverse pattern, scoring higher on PG (0.75) than on TMAS (0.64). In the appendices (Figure 10) we also show how the scores for these attributes vary depending on sequence length. We generally find that "clothing" and "age" remain highest regardless of sequence length, while "role" and "personality" decline as $n$ increases, suggesting abstract traits become harder to convey consistently as the task scales up. "Occupation" is an exception, improving monotonically with sequence length.

**Visual nature of attributes and plot relevance.** In Figure 7(a), we report EPICQA scores split into three bins by the visual nature (VN) of the underlying QA pairs (low: 3; medium: 4; high: 5). Both TMAS and PG show a positive trend: the high-visual scores highest (0.65 for TMAS, 0.64 for PG), while the low and medium bins score similarly at around 0.58. This result is intuitive: attributes that are more visually oriented are easier both for the illustration model to render faithfully and for the VQA model to verify. In Figure 7(b), plot relevance (PR) follows a more complex pattern. For TMAS, scores are highest at the lowest relevance level, dip across mid-range levels, before partially recovering at PR$=5$. For PG, the pattern is flatter, with slightly higher values at both extremes. On both datasets, mid-relevance attributes tend to score lower than those at the extremes, implying that such attributes occupy an ambiguous middle ground: they matter enough to be specified in the story but are not so central that the model is compelled to portray them consistently, nor so incidental that they are straightforward to depict.

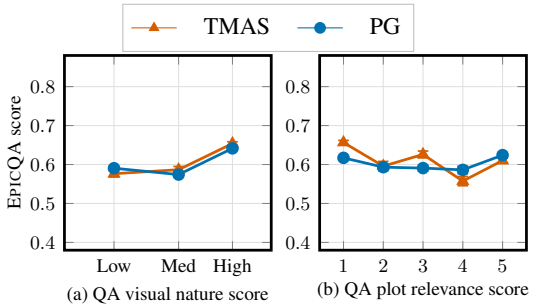


Figure 7: EPICQA performance for **Gemini 2.5 Flash** as a function of quality dimensions. Error bars show 95% confidence intervals. **(a) Visual nature:** Higher visualness scores correlate with higher EPICQA scores on both datasets. **(b) Plot relevance:** Both datasets show lower scores at mid-relevance levels.

## 8 Conclusion

We introduce the EPICQA evaluation framework for diagnosing story illustration models for their sensitivity to illustrating fine-grained character attributes. EPICQA pairs stories with counterfactual variants and uses automatically-generated Q/A to inspect whether illustration models are both sensitive to counterfactual changes in attributes as well as preserving details from the original. We use EPICQA to benchmark five state-of-the-art image generation models on this task and provide insights on how performance varies along dimensions such as sequence length and attribute type.

## Ethical Considerations

The task of story illustration is one where issues of hallucination, biases, and generating memorized content can have profound ethical implications. Our evaluation is meant to help with addressing these issues by measuring the robustness and sensitivity of the output images in their portrayal different attributes. However, realistic downstream usecases should perform more comprehensive analysis to understand how to mitigate these issues and be aware of the limitations of our proposed metric. One such limitation is that we only use English stories and have not tested how robust this method is when extending to other languages, which is important for fully understanding how these models can be used responsibly. Furthermore, we make use of stories from existing datasets which may carry their own biases. Project Gutenberg, although widely used in the NLP field, contains many sources of literature that are over a hundred years old, and so many stories contain antiquated language or reflect outdated values. Aside from the ethical considerations above, we also wish to disclose our usage of human annotations, for which we have paid crowdsource workers. Workers were informed of the task criteria and had to consent to the task. All workers are fluent English speakers, who were paid fairly, above minimum wage for their respective locations.

## Limitations

We limit our study to the story domain consisting of English stories, which means that more work would need to be done to extend this metric to other languages and confirm that the findings hold the same. We measure the performance on a set of about 2500 question pairs, but we only used 100 stories in our seed set, and this could be scaled up to a much broader set of initial narratives. Additionally, because gold truth images do not exist for these stories, we are limited to using metrics that do not compare to a gold truth. Lastly, the evaluation framework we introduce uses an LLM both for automatically creating counterfactual stories and then for scoring the QA pairs. To ensure quality control, we validated the quality in our paper via expert annotations (for the counterfactual creation) and crowdsourcers (for the QA scoring). However, this presents a limitation of the evaluation framework as it will always only be as reliable as the underlying LLM used for these intermediate tasks. This limitation should be kept in mind by those using this metric for downstream tasks especially on differing domains where the LLM quality may not transfer.

## References


Kabir Ahuja, Melanie Sclar, and Yulia Tsvetkov. 2025. Finding flawed fictions: Evaluating complex reasoning in language models via plot hole detection. In *Proceedings of COLM*.

Black Forest Labs. 2025. FLUX.2: Frontier visual intelligence. https://bfl.ai/blog/flux-2. Accessed: 2026.

Nils Dycke and Iryna Gurevych. 2026. Automatic reviewers fail to detect faulty reasoning in research papers: A new counterfactual evaluation framework. *Transactions of the Association for Computational Linguistics*, 14:465–488.

Matan Eyal, Tal Baumel, and Michael Elhadad. 2019. Question answering as an automatic evaluation metric for news article summarization. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3938–3948, Minneapolis, Minnesota. Association for Computational Linguistics.

Silin Gao, Sheryl Mathew, Li Mi, Sepideh Mamooler, Mengjie Zhao, Hiromi Wakaki, Yuki Mitsufuji, Syrielle Montariol, and Antoine Bosselut. 2025. Vinabench: Benchmark for faithful and consistent visual narratives. In *Proceedings of CVPR*.

Sayontan Ghosh, Mahnaz Koupaee, Isabella Chen, Francis Ferraro, Nathanael Chambers, and Niranjan Balasubramanian. 2023. Pasta: A dataset for modeling participant states in narratives. *Transactions of the Association for Computational Linguistics*, 11:1283–1300.

Google DeepMind. 2025a. Gemini 2.5 flash image model card. https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-2-5-Flash-Model-Card.pdf.

Google DeepMind. 2025b. Gemini 3 pro image. https://ai.google.dev/gemini-api/docs/models.

Tanmay Gupta, Dustin Schwenk, Ali Farhadi, Derek Hoiem, and Aniruddha Kembhavi. 2018. Imagine this! scripts to compositions to videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 598–613.

Changying Hao, Liang Pang, Yanyan Lan, Yan Wang, Jiafeng Guo, and Xueqi Cheng. 2021. Sketch and customize: A counterfactual story generator. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12955–12962.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceed-*

*ings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xudong Hong, Asad Sayeed, Khushboo Mehra, Vera Demberg, and Bernt Schiele. 2023. Visual writing prompts: Character-grounded story generation with curated image sequences. *Transactions of the Association for Computational Linguistics*, 11:565–581.

Ting-Hao Kenneth Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, C. Lawrence Zitnick, Devi Parikh, Lucy Vanderwende, Michel Galley, and Margaret Mitchell. 2016. Visual storytelling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1233–1239, San Diego, California. Association for Computational Linguistics.

Fantine Huot, Reinald Kim Amplayo, Jennimaria Palomaki, Alice Shoshana Jakobovits, Elizabeth Clark, and Mirella Lapata. 2025. Agents' room: Narrative generation through multi-step collaboration. In *The Thirteenth International Conference on Learning Representations*.

Yitong Li, Zhe Gan, Yelong Shen, Jingjing Liu, Yu Cheng, Yuexin Wu, Lawrence Carin, David Carlson, and Jianfeng Gao. 2019. StoryGAN: A sequential conditional GAN for story visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6329–6338.

Chang Liu, Haoning Wu, Yujie Zhong, Xiaoyun Zhang, Yanfeng Wang, and Weidi Xie. 2024. Intelligent Grimm – open-ended visual storytelling via latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6190–6200.

Adyasha Maharana and Mohit Bansal. 2021. Integrating visuospatial, linguistic, and commonsense structure into story visualization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6772–6786, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Feiteng Mu and Wenjie Li. 2024a. A causal approach for counterfactual reasoning in narratives. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6556–6569, Bangkok, Thailand. Association for Computational Linguistics.

Feiteng Mu and Wenjie Li. 2024b. Generating contrastive narratives using the brownian bridge process for narrative coherence learning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6538–6555, Bangkok, Thailand. Association for Computational Linguistics.

OpenAI. 2025. GPT Image 1.5. `https://developers.openai.com/blog/openai-for-developers-2025`.

Lianhui Qin, Antoine Bosselut, Ari Holtzman, Chandra Bhagavatula, Elizabeth Clark, and Yejin Choi. 2019. Counterfactual story reasoning and generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5043–5053, Hong Kong, China. Association for Computational Linguistics.

Jack W Rae, Anna Potapenko, Siddhant M Jayakumar, Chloe Hillier, and Timothy P Lillicrap. 2019. Compressive transformers for long-range sequence modelling. *arXiv preprint*.

Tanzila Rahman, Hsin-Ying Lee, Jian Ren, Sergey Tulyakov, Shweta Mahajan, and Leonid Sigal. 2023. Make-a-story: Visual memory conditioned consistent story generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2493–2502.

Yoad Tewel, Omri Kaduri, Rinon Gal, Yoni Kasten, Lior Wolf, Gal Chechik, and Yuval Atzmon. 2024. Training-free consistent text-to-image generation. *ACM Transactions on Graphics*, 43(4). SIGGRAPH 2024 journal track.

Xiyu Wang, Yufei Wang, Satoshi Tsutsui, Weisi Lin, Bihan Wen, and Alex C. Kot. 2024. Evolving storytelling: Benchmarks and methods for new character customization with diffusion models.

Olivia Wiles, Chuhan Zhang, Isabela Albuquerque, Ivana Kajić, Su Wang, Emanuele Bugliarello, Yasumasa Onoe, Chris Knutsen, Cyrus Rashtchian, Jordi Pont-Tuset, and Aida Nematzadeh. 2025. Revisiting text-to-image evaluation with gecko: On metrics, prompts, and human ratings. In *Proceedings of ICLR*.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, and 20 others. 2025. Qwen-image technical report. *Preprint*, arXiv:2508.02324.

Te-Lin Wu, Zi-Yi Dou, Qingyuan Hu, Yu Hou, Nischal Chandra, Marjorie Freedman, Ralph Weischedel, and Nanyun Peng. 2023. ACQUIRED: A dataset for answering counterfactual questions in real-life videos. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11753–11770, Singapore. Association for Computational Linguistics.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. 2024. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1819–1862, Mexico City, Mexico. Association for Computational Linguistics.

Shuai Yang, Yuying Ge, Yang Li, Yukang Chen, Yixiao

Ge, Ying Shan, and Ying-Cong Chen. 2025. Seed-story: Multimodal long story generation with large language model. pages 1871–1881.

Xi Ye, Rohan Nair, and Greg Durrett. 2021. Connecting attributions and QA model behavior on realistic counterfactuals. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5496–5512, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zilyu Ye, Jinxiu Liu, Ruotian Peng, Jinjin Cao, Zhiyang Chen, Yiyang Zhang, Ziwei Xuan, Mingyuan Zhou, Xiaoqian Shen, Mohamed Elhoseiny, Qi Liu, and Guo-Jun Qi. 2024. Openstory++: A large-scale dataset and benchmark for instance-aware open-domain visual storytelling. *arXiv preprint arXiv:2408.03695*.

Yupeng Zhou, Daquan Zhou, Ming-Ming Cheng, Jiashi Feng, and Qibin Hou. 2024. StoryDiffusion: Consistent self-attention for long-range image and video generation. In *Advances in Neural Information Processing Systems*, volume 37. NeurIPS 2024 Spotlight.

Cailin Zhuang, Ailin Huang, Yaoqi Hu, Jingwei Wu, Wei Cheng, Jiaqi Liao, Hongyuan Wang, Xinyao Liao, Weiwei Cai, Hengyuan Xu, Xuanyang Zhang, Xianfang Zeng, Zhewei Huang, Gang Yu, and Chi Zhang. 2026. Vistorybench: Comprehensive benchmark suite for story visualization. In *Proceedings of CVPR (to appear)*.

## A Story Data Access

We use 50 stories each from the Tell-Me-A-Story (Huot et al., 2025) and Project Gutenberg 19 (Rae et al., 2019) datasets. Both of these datasets are publicly available via Github and released under the Apache License 2.0. We select stories from the training set of each and filter out stories with too few visually relevant character attributes ($< 4$). For Project Gutenberg, we cross-referenced with the index available on `gutenberg.org` to filter only stories in English that have been tagged with "fiction" labels. We split the Project Gutenberg story into chapters automatically with Gemini 2.5 Flash, and only use the first chapter of each. Additionally, to limit context overflow issues, we truncate all stories to be up to 5000 word tokens.

## B Model Implementation Details

We use multiple LLMs in this paper, all of which are used with their publicly available packages or APIs with the default settings, unless stated otherwise. For the counterfactual creation pipeline, we use standard text decoding settings, with a temperature $= 1.0$ in most stages. However, in Stage 2 (alternate attribute generation), we increase the temperature to 1.5 to increase the diversity of the counterfactual attribute values.

For the image generation models, we mostly use them with out-of-box settings. We did have a few issues with certain model families (GPT and Flux) being unable to generate multiple images per story even when prompted. In practice, the only way to address this was to split the story into 3 or 5 segments and prompt these models to illustrate each part separately. The prompts given to each model were otherwise the same. The prompt template for the image generation models was of the format: "Create x pictures to illustrate x key scenes from the following story." where $x \in \{1, 3, 5\}$

## C EpicQA Dataset Generation Prompts

This section documents the five zero-shot prompts used by the counterfactual data-generation pipeline of Section 4. Throughout, placeholders of the form `{name}` are replaced with the corresponding input at runtime. Prompts are presented in pipeline order: character and attribute extraction (Stage 1), alternate attribute generation (Stage 2), counterfactual rewriting (Stage 3) using prompts adapted from Ahuja et al. (2025)), and attribute scoring

along plot relevance and visual nature (Stage 4). We removed all attributes of type "action" that the pipeline produced as these tended to be descriptions of dynamic events in the story that may span multiple images and be difficult to evaluate. Also, the counterfactual story output in Stage 3 has <m> tags around the rewritten portions of the text. We *remove these tags* before using the counterfactual stories in the image generation and evaluation setups in order to avoid unfairly hinting to the models which sections of the text to focus on.

**Stage 1: Character and Attribute Extraction**

I'm trying to write a counterfactual version of the following story that will be illustrated by another model, and I need help finding visual character attributes that can be changed without affecting the plot.

A counterfactual story should follow a similar chain of events as the original but names, visual traits, or roles could be changed throughout the story.

Please help me by reading the story below and identify the main characters and a set of attributes that describe them.

First identify the main characters (1–3 entities who drive forward the plot of the story). Each character should be a single entity. List as many characters as possible.

Then select a list of attributes that describe each character. An attribute is any type of character trait, including the character's name, gender, nationality, hair color, age, clothes, etc. Attributes should be identifiers that would distinguish this character. Attribute values should be described in 1–3 words. Please list as many attributes as possible, but only pick attributes that can be described in a few words.

**Story**

{story_text}

**Output Format**

A separate bulleted list of attributes for each character. Use * for each bullet point. Separate each character list with a double newline.

*Example output format:*

```
Character: <Character Name>
* <Attribute type>: <Attribute Value>
* <Attribute type>: <Attribute Value>
* <Attribute type>: <Attribute Value>
* <Attribute type>: <Attribute Value>

Character: <Character Name>
* <Attribute type>: <Attribute Value>
* <Attribute type>: <Attribute Value>
* <Attribute type>: <Attribute Value>

Character: <Character Name>
* <Attribute type>: <Attribute Value>
* <Attribute type>: <Attribute Value>
* <Attribute type>: <Attribute Value>
```

**Stage 1 (cont.)**

Follow the above format exactly.

**Stage 2: Alternate Attribute Generation**

I'm trying to write a counterfactual version of the following story that will be illustrated by another model, and I need help finding visual character attributes that can be changed without affecting the plot. A counterfactual story should follow a similar chain of events as the original but names, visual traits, or roles could be changed throughout the story.

**Story**

{story_text}

The character {char} in this story has the following attribute:

**Attribute**

{attr_key}: {attr_value}

Can you imagine how the story would be different if the attribute was changed? First come up with a different value for the attribute above. Make sure the new value is *noticeably different* from the original but still plausible.

**Output Format**

New attribute value: <Attribute Value>

**Stage 3: Counterfactual Story Rewriting**

Consider the story below:

**Story**

{story}

In this story it is established that the character "{character}" has a "{att_type}" that is "{attval}". What if instead their {att_type} was "{cfval}"? Can you re-write the story considering this what-if scenario? Try to stick close to the original plot but do make the necessary changes which would arise naturally on altering this fact. Note that if there are multiple possibilities for altering a fact, then choose the one which results in minimal changes to the original story. The modified story should appear natural and feel it was written with the flipped fact as the original intent. Avoid stating the flipped fact as a simple negation of the fact and have it implied instead. Mark each line which was modified as a result of this change to be enclosed in the tags <m></m>. First start by brainstorming what changes would result on flipping the fact, followed by the altered story with the fact flipped.

**Output Format**

**Stage 3 (cont.)**

```
## Brainstorming
<Reasoning about the changes in the story due to flipping of the fact>

## Counterfactual Story
<New Story>
```

**Stage 4a: Plot-Relevance Scoring**

Please read the story and character attribute below. For the given attribute below assign a score (between 1 and 5) of how much the attribute affects the character's behavior and personality.

**Story**

{story_text}

**Character**

{charname}

**Attribute**

{attkey}: {attvalue}

**Output Format**

Score: <Score 1-5>

### C.1 Stage 4b: Visual-Nature Scoring

**Stage 4b: Visual-Nature Scoring**

Please read the story and character attribute below. For the given attribute below assign a score (between 1 and 5) of how **visual** the attribute is.

**Story**

{story_text}

**Character**

{charname}

**Attribute**

{attkey}: {attvalue}

**Output Format**

Score: <Score 1-5>

## D Adding Bounding Boxes

Prior to asking the VQA model any of the attribute based questions in EPICQA, we automatically create a bounding box around the character being referred to in the question and draw it directly on the image. The purpose of this is two-fold. First, it takes care of entity coreference in multi-character images. Second, it allows us to remove the character's name from the question altogether. This is important because sometimes the name contains clues for answering the rest of the question (e.g. Does Dr. Lydia Brooks have an occupation that is a doctor? Does Dr. Lydia Brooks have a gender that is female?). To add the bounding boxes, we have an extra pre-processing step where we use Gemini 2.5 Flash to produce the x and y coordinates of a box enclosing the character in an image or output n/a if the character doesn't appear in the image. The bounding box generation prompt provides the model with the story text, image, and character name. If the bounding box that is generated is a set of x and y coordinates, we add a red rectangle to the image, and this new image is what's used for the VQA step. If the model said N/A (the character doesn't appear in the image), we skip using the VQA model for that example and simply output N/A to the question.

## E LLM Autorater Prompt

The following prompt is used to elicit holistic quality ratings from the LLM autorater (Gemini). The rubric component (§E.1) is concatenated with the format instructions (§E.2) before being passed to the model alongside the story prompt and generated illustrations.

### E.1 Scoring Rubric

**System / Rubric Prompt**

You are an expert art director and story editor. Your task is to critically evaluate a sequence of illustrations for a story based on a detailed rubric.

You will be given a story prompt and a sequence of generated illustrations (which will contain 1, 3, or 5 images). The images are presented in sequential order (Image 1, Image 2, etc.), corresponding chronologically to the narrative progression of the story.

Your response must be structured precisely. For each of the 5 dimensions defined below, you must provide a score from 1 to 5 and a brief, specific justification for your score. Your justification must explicitly reference the scoring criteria and cite examples from the provided inputs. After scoring all dimensions, calculate and state the overall average score as a decimal number (e.g., 3.8).

**Dimension 1: Stylistic Cohesion**

*Measure the consistency of the artistic style across the sequence of illustrations. Evaluate elements like color palette, lighting, line work, and texture rendering. This*

**System / Rubric Prompt (cont.)**

*includes both the broad style (e.g., anime, watercolor) and more subtle qualities that create a unified look. (Note: If only 1 image is provided, evaluate its internal stylistic consistency.)*

- **Score 5** (*Excellent*): All illustrations share a single, coherent artistic style. The color palette, lighting model, line work, texture detail, and rendering techniques are uniform and consistent.
- **Score 4** (*Very Good*): The style is highly consistent with only negligible, hard-to-spot deviations.
- **Score 3** (*Average*): The style is mostly consistent, but some images have noticeable deviations in secondary elements (e.g., different lighting temperature, or noticeably sharper/softer texture rendering).
- **Score 2** (*Below Average*): Clear and distracting differences in core style between multiple images (e.g., one is painterly, another looks like a 3D render).
- **Score 1** (*Poor*): The sequence looks like a random assortment of images from different artists with fundamentally different styles and palettes.

**Dimension 2: Narrative Fidelity and Thematic Alignment**

*Evaluate how well the illustrations capture the provided story prompt and depict its progression in a logical sequence. The generation of the expected number of images (whether 1, 3, or 5) is critical for narrative completeness.*

- **Score 5** (*Excellent*): The narrative arc is clearly and unambiguously depicted in the correct order. The illustrations accurately reflect the story prompt and effectively translate the text into a visual narrative.
- **Score 4** (*Very Good*): The main narrative elements are depicted, but the progression might be slightly ambiguous or one scene less clear than the others.
- **Score 3** (*Average*): The story is only partially conveyed. Important narrative beats are missing, ambiguous, or misinterpreted. The overall story flow feels disjointed.
- **Score 2** (*Below Average*): The visual narrative fails to capture most of the story. Also applies if fewer images than the expected sequence length are generated.
- **Score 1** (*Poor*): The images fail entirely to depict the story. **This score must be given if fewer than the total of expected images are generated.** The illustrations seem disconnected from the story prompt.

**Dimension 3: Composition and Visual Interest**

*Assess the artistic merit, including composition, dynamism, and diversity across the sequence. Also includes checking for logical and compositional failures within individual frames (e.g., a character being duplicated incorrectly within the same image).*

- **Score 5** (*Excellent*): The sequence is visually diverse and compelling, free of compositional errors, employing a variety of camera angles, framing, and dynamic, emotionally resonant poses.
- **Score 4** (*Very Good*): Compositions are strong and varied, with only minor moments of repetition or uninspired framing.
- **Score 3** (*Average*): Compositions are competent but somewhat repetitive or generic (e.g., all simple medium shots), lacking dynamism.
- **Score 2** (*Below Average*): Most compositions are weak, awkward, or repetitive, or a scene contains a significant logical failure (e.g., the same character appearing twice in a single frame).
- **Score 1** (*Poor*): Illustrations are compositionally awkward or static. **Also applies if there are major logical failures such as multiple instances of the main character in the same image.**

**Dimension 4: Technical Quality and Artifacts**

*Perform a technical check for flaws, glitches, and other common AI-generation artifacts across all images.*

- **Score 5** (*Excellent*): Images are clean, sharp, and free of noticeable artifacts. Anatomy (hands, faces) is correct, objects are coherent, and there is no strange blending, texture warping, or nonsensical elements.
- **Score 4** (*Very Good*): Very minor, hard-to-spot artifacts present that do not detract from overall image quality.
- **Score 3** (*Average*): Minor but noticeable artifacts present (e.g., slightly misshapen hands, distorted background faces, minor texture errors) that are clearly visible on inspection.
- **Score 2** (*Below Average*): Multiple distracting artifacts that negatively impact the viewing experience.
- **Score 1** (*Poor*): Images are plagued with major, distracting artifacts. Key characters or objects have significant anatomical or logical errors (e.g., extra limbs, distorted face). Images are blurry or incoherent.

**Dimension 5: Prompt-Level Detail Adherence**

*Assess how faithfully the illustrations follow the specific, detailed elements mentioned in the story prompt, including character descriptions, specific environmental elements, actions, or aesthetics explicitly requested.*

- **Score 5** (*Excellent*): The illustrations are perfect or near-perfect visual executions of the story prompt's specific details. All key directives (actions, character features, specified objects, settings) are correctly and clearly rendered.
- **Score 4** (*Very Good*): Highly faithful to prompt details but misses or alters one minor element that does not significantly impact the core meaning of the scene.
- **Score 3** (*Average*): Captures the general subject of the prompt but ignores or fails to render several key descriptive details (e.g., missed aesthetic choice, wrong character clothing, absent requested object).

**System / Rubric Prompt (cont.)**

- **Score 2** (*Below Average*): Only loosely relates to the prompt, capturing the broad theme but ignoring specified actions, descriptions, and most other details.
- **Score 1** (*Poor*): Little or no connection to the prompt details, resulting in images that seem random or entirely unrelated to the specific instructions given.

**Overall Score:** Average of scores across all 5 dimensions.

### E.2 Format Instructions

**Format Instructions**

You **must** indicate your answer in the following JSON format. Omit any text before the beginning of the answer. Omit the word `json` at the beginning of the answer.

```
{
  "story_illustration_score": [
    {
      "rating_dimension": "Stylistic
          Cohesion",
      "score": x,
      "justification": "
          justification"
    },
    {
      "rating_dimension": "Narrative
          Fidelity and Thematic
          Alignment",
      "score": x,
      "justification": "
          justification"
    },
    {
      "rating_dimension": "
          Composition and Visual
          Interest",
      "score": x,
      "justification": "
          justification"
    },
    {
      "rating_dimension": "Technical
          Quality and Artifacts",
      "score": x,
      "justification": "
          justification"
    },
    {
      "rating_dimension": "Prompt-
          Level Detail Adherence",
      "score": x,
      "justification": "
          justification"
    },
    {
      "rating_dimension": "Overall
          Score",
      "score": x,
      "justification": "
          justification"
    }
  ]
}
```

## F Additional Results on Illustration Quality

In Figure 9 we report extended analysis on Gemini-2.5-Flash outputs beyond what's in the main paper (expanded from Figure 7). As a comparison point, we include Figure 8 which reports the same four complexity analyses for images generated by Gemini 3 Pro, allowing a direct comparison with Gemini 2.5 Flash to check that patterns hold similarly across different models. We find that the broad patterns are largely consistent across the two models, but a few differences emerge.

**Story input length.** We investigate how the input length of the story might affect the EpicQA scores in Figure 9(a). We observe broadly opposite trends on TMAS and PG data. For TMAS, longer input stories correlate with higher EpicQA scores while for PG shorter stories tend to score higher, though with some non-monotonic variation at intermediate lengths. One possible explanation could be differences in information density across the two domains, meaning that more tokens are more helpful or obfuscating in different domains. Another possible explanation could be differences in memorization biases: longer PG stories are more likely to have been extensively seen at pretraining time, making it harder for the model to adapt its depictions to the specific attributes described in the text rather than defaulting to memorized representations. For TMAS stories, which are unlikely to have been memorized, additional length may instead provide richer context that helps the model generate more faithful illustrations.

Looking at EpicQA scores of Gemini-Pro images for comparison (Figure 8), results show the same divergent TMAS/PG trend: TMAS scores increase with token length while PG scores decline from a high of around 0.69–0.71 for the shortest stories. The magnitudes are very similar, with Gemini Pro tracking Gemini Flash closely across all length bins on both datasets. The PG decline is consistent for both models.

**Reading complexity.** As a proxy for story complexity, we use the Flesch–Kincaid grade level (see Figure 9(b)), acknowledging that readability met-

rics capture only syntactic and lexical difficulty rather than narrative complexity in a broader sense. The effect is weaker and less consistent than that of token length. The confidence intervals are tight given the large number of questions per bin, and the overall range of variation is small (within approximately 0.04 points for both datasets), suggesting that syntactic and lexical complexity, as captured by readability metrics, is not a primary driver of illustration quality, the domain and memorization status of the story appear to matter considerably more. Gemini-Pro and Gemini-Flash are again closely matched, though Pro shows a slightly flatter profile on TMAS, with a less pronounced dip at grades 5–7 (0.62 for Pro vs. 0.61 for Flash). On PG, both models show a similar pattern of low scores at mid-range grades and a recovery at the highest grade levels.

**Visual nature.** The positive relationship between attribute visualness and EPICQA score is preserved for Gemini-Pro on both datasets, consistent with Flash. One notable difference is that Pro shows a more pronounced step from the medium to the high visualness bin on TMAS (0.616 to 0.654), whereas Flash shows a more gradual increase across all three bins. On PG, the two models are nearly indistinguishable.

**Plot relevance.** The pattern for Gemini-Pro closely mirrors that of Gemini-Flash on both datasets: scores are highest at the lowest relevance level ($\text{PR} = 1$), dip across mid-range levels, reach a minimum at $\text{PR} = 4$, and partially recover at $\text{PR} = 5$. This consistency across both models suggests the pattern reflects a genuine property of mid-relevance attributes being ambiguous to depict and evaluate, rather than an artifact of any particular model's behaviour.

## G Gecko Q/A in more depth

We compare to Gecko, an image evaluation framework that generates questions from the input text and scores the illustration based on the VQA scores of the questions. However, Gecko off-the-shelf performance is consistently below random on our story illustration domain. We observe a few reasons why Gecko scores are so low on this dataset, attributing it mostly to Gecko being intended for use with short input texts (like image captions) which are more literal descriptions of concretely what's in the image. Gecko failures seem to fall in three categories: (a) Gecko's Q/A filtering being poorly adapted to the narrative setting and (b) Gecko's Q/A generation prompt is intended for short captions and when using longer narratives it produces many visually irrelevant questions or entity-existence questions that are not as useful for this task set-up (c) unlike EPICQA, Gecko doesn't use a bounding box to identify characters in an image, which makes many questions unanswerable because they refer to named entities.

First, regarding filtering, we observe that for over half the stories, Gecko filters out all of the QA it generates before scoring the story, meaning it doesn't output any meaningful score. This may be because of issues with the NLI filtering in Gecko which is not designed for long text like the narratives we use as input. Because of this, we tried disabling the NLI-based filtering in Gecko, which increased the number of questions it used substantially, so that every story now has questions used in the scoring. But removing the filtering doesn't affect the overall score by much. Overall, Gecko scores were still less than random chance.

We also observe that the questions generated by Gecko are typically asking about the presence of certain entities rather than more complex visual relationships. This makes sense because Gecko was developed around literal image captions, and in fact the in-context few-shot examples in Gecko's Q/A generation prompt prime for these type of questions. This type of "entity-existence" question is reasonable in the captioning setting but less useful for aligning images to full-length narratives. Here is a cherry-picked set for reference of the types of questions generated by the Gecko framework about the existence of entities:

- Are rabbits mentioned?
- Is Kathleen a character in the story?
- Is there a table in the room?
- Is a valley described in the story?
- Is there a metallic sphere?
- Is there a bottle of soda?
- Are there hanging shelves?

Many of the Gecko Q/A also mention named-entities which are inherently unanswerable from the image without extra coreference information to map to the correct character in the image. Our

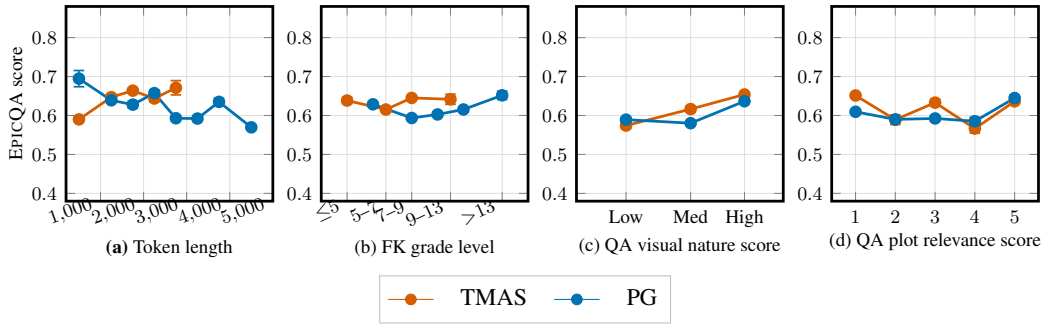


Figure 8: EpicQA performance for **Gemini 3 Pro** as a function of four complexity and quality dimensions computed over all questions pooled across all sequence lengths. Error bars show 95% confidence intervals. **(a) Story length:** TMAS scores increase with token length while PG shows the opposite trend, consistent with memorization of longer canonical texts. **(b) Reading complexity:** Both datasets show a non-monotonic pattern across Flesch–Kincaid grade levels, with a dip at mid-range complexity. **(c) Visual nature:** Higher visualness scores correlate with higher EpicQA scores on both datasets. **(d) Plot relevance:** Both datasets show lower scores at mid-relevance levels, consistent with these being the most ambiguous attributes to depict.

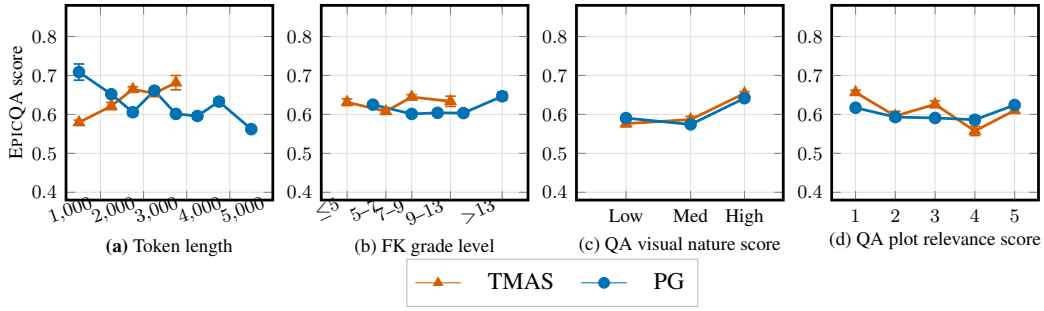


Figure 9: EpicQA performance for **Gemini 2.5 Flash** as a function of four complexity and quality dimensions. Error bars show 95% confidence intervals. **(a) Story length:** TMAS scores increase with token length while PG shows the opposite trend. **(b) Reading complexity:** Both datasets show a non-monotonic pattern across Flesch-Kincaid grade levels, with a dip at mid-range complexity. **(c) Visual nature:** Higher visualness scores correlate with higher EpicQA scores on both datasets. **(d) Plot relevance:** Both datasets show lower scores at mid-relevance levels, consistent with these being the most ambiguous attributes to depict.

framework, EpicQA, gets around this issue by adding bounding boxes around the character the question is about, but this type of coreference is not built into Gecko's design and would be difficult to add. Additionally many questions are about story elements that are irrelevant to the visuals (e.g. Is William Penn mentioned in relation to Pennsylvania?).

## H CLIP Scores

We also computed CLIP scores (Hessel et al., 2021) as a measure of the similarity between the input story and the images. CLIP requires a maximum context length which is much shorter than our full length stories. In order to use CLIP, we have to first automatically summarize the story, though this is not ideal as it means a significant loss of information. We report the results in Figure 11. Scores are generally low but with a clear separation between proprietary and open-source models.

## I Human Evaluation Instructions

### I.1 Expert Evaluation of the Counterfactual Pipeline

Expert annotators were given an HTML table with each row containing the original story, the target attribute, its assigned counterfactual value, and the rewritten story.

Annotators were asked to label:

- Does the counterfactual story make sense? **Yes / No** (Is the rewritten story coherent and logical?)
- Is the counterfactual information fully incorporated everywhere? **Yes fully / Partially /**

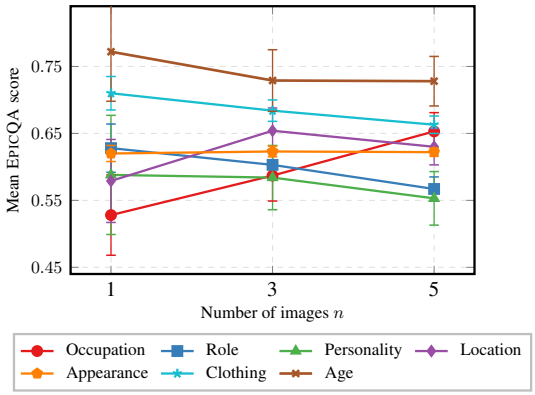


Figure 10: Mean EpicQA consistency score per attribute type for **Gemini 2.5 Flash** as a function of illustration sequence length $n$, aggregated across TMAS and PG. Occupation is the only attribute that improves monotonically with $n$; role and personality decline. Appearance is the most stable attribute across sequence lengths.

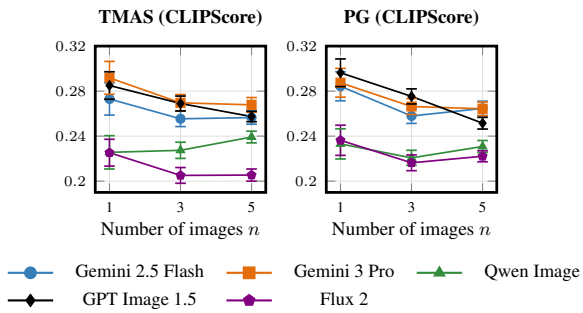


Figure 11: CLIP metrics scores as a function of illustration sequence length $n \in \{1, 3, 5\}$. Col 1: TMAS; Col 2: PG.. Error bars show 95% confidence intervals.

**No** (Was the attribute change applied consistently throughout the story?)

- Did the counterfactual story only make the minimal necessary changes? **Yes / No** (Did the rewrite avoid changing things that didn't need to change?)

### I.2 Crowdsourced evaluation of VQA Quality

Annotators were given the following instructions:

Look at the image and read the question above the image. The question should be a yes/no question asking about one of the characters (people or animals) who has a red box drawn around them (for example: "Does the character in the following bounding box have red hair?").
Look at the image below and try to answer the question (either Yes or No). If the question cannot be answered by the image (perhaps it asks about something that is not shown in the image), then select "cannot tell from the image". If there are no characters in the bounding box, select "No character in the bounding box".
After answering the question, rate how confident you are in your answer.
*If the bounding box is empty*: select "No character in the bounding box".
*If the bounding box only partially contains a character*: Try your best to answer the question about the character who is partially in the bounding box.
*If the question is grammatically incorrect or does not make sense*: Don't worry if the grammar of the question is not perfect. If you can understand the overall meaning of the question, just try your best to answer if possible, but if you cannot understand it at all, then select "cannot tell from the image".