Generating Visual Stories with Grounded and Coreferent Characters

Danyang Liu, Mirella Lapata, Frank Keller
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh EH8 9AB, UK
danyang.liu@ed.ac.uk, {mlap, keller}@inf.ed.ac.uk

Abstract

Characters are important in narratives. They move the plot forward, create emotional connections, and embody the story's themes. Visual storytelling methods focus more on the plot and events relating to it, without building the narrative around specific characters. As a result, the generated stories feel generic, with character mentions being absent, vague, or incorrect. To mitigate these issues, we introduce a new character-centric approach to visual story generation. We present the first model capable of predicting visual stories with consistently grounded and coreferent character mentions. Our model is finetuned on a new dataset which we build on top of the widely used VIST (Huang et al., 2016) benchmark. Specifically, we develop an automated pipeline to enrich VIST with visual and textual character coreference chains. We also propose new evaluation metrics to measure the richness of characters and coreference in stories. Experimental results show that our model generates stories with recurring characters which are consistent and coreferent to larger extent compared to baselines and state-of-the-art systems.[1]

1 Introduction

An integral part of storytelling is creating interesting and believable characters. In fact, it is not uncommon for writers to conceptualize their characters visually before crafting a plot around them. This character-centric approach enhances narrative coherence, leading to richer, engaging, and emotionally resonant stories. Recent research has explored how character-centric datasets (Porvatov et al., 2024; Chen et al., 2023; Brahman et al., 2021) as well as different representations of characters might contribute to the automatic analysis and generation of narratives (Gurung and Lapata, 2024; Inoue et al., 2022; Bamman et al., 2013; Kim et al., 2018a; Liu et al., 2020; Brahman et al., 2021; Chen et al., 2024; Li et al., 2024; Xu et al., 2024; Yu et al., 2022; Yang et al., 2023b, 2022).

However, the importance of characters seems to go unnoticed in narrative tasks spanning multiple modalities such as visual storytelling, which involves narrating a story based on a sequence of images. Existing approaches focus on detecting objects in images and discovering relationships between them; characters are treated the same as other objects, without any special consideration in the generation process. For example, the most popular approaches to visual storytelling (Wang et al., 2024; Chen et al., 2021; Hsu et al., 2020; Yang et al., 2019) exploit external knowledge bases to enrich and link detected concepts. While such methods can describe a coherent sequence of events, they fail to effectively ground character mentions to their visual depictions and they do not generate character-centric stories. Character mentions are often absent, vague (e.g., rendered with plural references such as ''they'' or ''we''), or incorrect, and the stories feel generic, lacking in detail and context. Figure 1 gives representative examples that illustrate how story quality depends on the presence of visually grounded characters.

It is not possible to create character-centric stories without knowing who the characters are. We therefore propose to identify them in an image sequence together with their visual coreference chains. In Figure 2, different characters are segmented using different colors, and are assigned a unique ID. Characters with the same ID in different images form a coreference chain. There are five characters in the image sequence (labels 1–5), and four coreference chains (characters 1 and 4 are shown in three images, character 2 is shown in two, while characters 3 and 5 are depicted only once). We would then expect a generation model that takes these annotations into account to

[1]Our code and dataset are available at https://github.com/iz2late/character-centric-vist.

Transactions of the Association for Computational Linguistics, vol. 14, pp. 442–464, 2026. https://doi.org/10.1162/TACL.a.641
Action Editor: Julia Hockenmaier. Submission batch: 9/2024; Revision batch: 7/2025; Published 4/2026.
© 2026 Association for Computational Linguistics. Distributed under a CC-BY 4.0 license.

Character Absence: There were a lot of fireworks that night. Some of them were single colored. Others were multi colored. Some of them were very bright. Others made huge explosions.

Character Vagueness: There was a parade going on. The first thing up was a fire truck. A man watched on. Later on there was a street party. It ended with a big fireworks show.

Incorrect Coreference: We all met up with the men. They walked a lot in the man walking. We had a lot of food. I gave a speech. Everyone was looking forawrd to it. Afterward, we all got together for pictures.

Figure 1: Examples from the VIST dataset illustrating how the absence of characters or vague character references affect story coherence and engagement. The first story is purely descriptive without any characters, lacking emotional depth and narrative engagement. No human presence means no perspective, making it static and impersonal. The second story mentions character, ‘‘a man’’, without, however, making the story about him. The character does not drive the narrative forward, and the story feels as uninspired as the first one. The third story does not refer to the protagonist correctly, switching between ‘‘we’’, ‘‘they’’, and ‘‘I’’, which makes the narrative nonsensical.

Input: Image Sequence with Visual Character Coreference Annotations

Output: Story with Textual Character Coreference Annotations

It is [Ella](#4)'s 11th birthday and [the family](#1, #2, #3, #4, #5) is all there to celebrate. [Ella](#4) looks at her cake waiting for the chance to blow out the candles. After [she](#4) has blown them out [she](#4) cuts the cake. [Her brother](#1) watches amused and excited that it is [Ella](#4)'s Birthday. [Ella's two younger siblings](#2,#5) watch as she(#4) opens gifts with some envy.

Figure 2: A sample from the VIST dataset (Huang et al., 2016) augmented with character chains. Visual characters are outlined by segmentation mask boundaries, where same color indicates same characters. Each character bears a unique label overlaid in the center of the mask. Output stories are annotated with textual character chains, and each character mention is aligned to visual characters (e.g., (#4) refers to visual segment 4).

refer to the same character consistently, creating textual coreference chains which are nevertheless grounded. Again, in Figure 2 ‘‘Ella’’ and ‘‘she’’ refer to visual character 4, whereas ‘‘Ella’s two younger siblings’’ refers to characters 2 and 5. Note that without forcing the model to explicitly ground character mentions in images, there is no way of knowing which visual characters the model is talking about.

In this work, we introduce a novel *character-centric* approach to visual story generation, which creates stories with character mentions grounded to the input image sequence. As explained earlier, this approach requires understanding coreference relationships between visual characters and aligning textual character mentions to their visual counterparts. This type of grounding is key to generating coherent and detailed stories, anchoring

textual mentions to specific visual appearances, and keeping the story closely linked to the image context.

Perhaps unsurprisingly, existing datasets do not contain annotations such as those shown in Figure 2. We augment the widely used VIST dataset (Huang et al., 2016) with visual and textual character annotations following a fully automated pipeline. We use OTTER-LLAMA-7B (Li et al., 2023), which has been instruction-tuned for multiple image understanding, as our backbone model and train it on image sequences and stories enriched with visual and textual coreference chains (see Figure 2). We render visual character chains into visual prompts and enforce character grounding in the generated stories via a new format which links character mentions to visual ids. We further introduce novel automatic metrics which assess character richness and coreference in stories, and propose to use LLMs as evaluators (LLM as a judge) to perform side-by-side comparisons of stories generated by different systems. The contributions of our work can be summarized as follows:

- We introduce a new, character-centric approach to story generation and present the first model capable of generating stories with consistent and grounded character mentions.
- Recognizing the lack of character annotations in visual stories, we enrich the VIST benchmark (Huang et al., 2016) with visual and textual character coreference chains and their corresponding alignment. The new benchmark, which we call VIST++, contains detailed annotations for 300K unique characters over 40K visual stories.
- We propose new evaluation methods to measure the richness of characters and coreference in stories. We further replace costly human evaluation with an LLM-as-a-Judge approach, which we use to compare visual stories along various dimensions (e.g., specificity, coherence, grounding).

2 Character-centric Visual Story Generation

Our work aims to address limitations of current visual storytelling systems, which struggle to accurately recognize character coreference relationships in the input images. Different from traditional visual storytelling tasks, where the input is an image sequence and output is a story, our character-centric approach takes image sequences and their corresponding visual character coreference chains as input, and produces a story with grounded and coreferring characters as output. By *explicitly* linking textual characters to their visual counterparts, we ensure character consistency, reduce vague or incorrect references, and facilitate the generation of more accurate, detailed, and engaging narratives. In the following, we first discuss how we automatically augment VIST (Huang et al., 2016) with character chains (Section 2.1) and then present our visual story generation model (Section 2.2).

2.1 The VIST++ Dataset

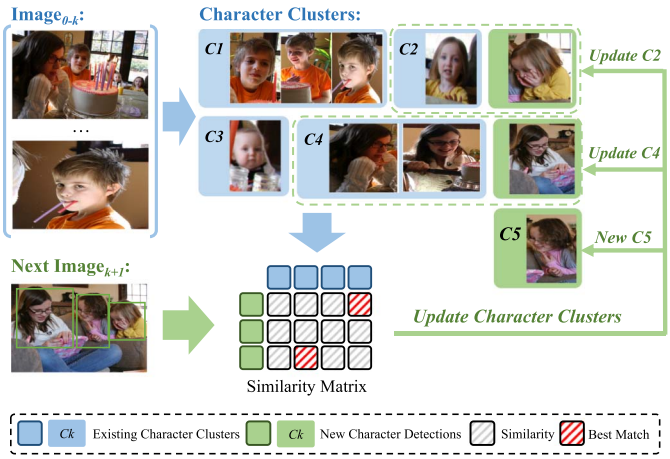
We developed an automated pipeline to enrich VIST (Huang et al., 2016) with character annotations, including fine-grained character labels in images and aligned textual coreference chains. The resulting VIST++ dataset contains 40K visual stories, with 300K unique characters, each associated with segmentation masks across images and textual coreference chains, amounting to a total of 520K character appearances.

Our pipeline includes three subtasks: (1) *visual character coreference*, i.e., identifying characters in the image sequence and grouping those that are the same person into coreference chains; (2) *textual character coreference* involves detecting character mentions and identifying coreference chains in the story text; and (3) *multimodal alignment* links textual to visual coreference chains, yielding multi-modal chains. We next introduce our automatic procedure for each task.

2.1.1 Visual Character Coreference

To obtain visual character chains, we first detect characters in an image sequence and then identify which detections represent the same character, thereby constructing coreference chains.

Previous visual coreference methods (Schroff et al., 2015; Liu and Keller, 2023) group bounding boxes (e.g., based on facial features) directly into a fixed number of clusters which should ideally vary from image to image. Our approach differs in two respects. Firstly, we use segmentation masks rather than bounding boxes; the latter are well suited to rectangular shapes, but are less accurate with objects that have irregular boundaries. Moreover, segmentation masks can distinguish



(a) Visual Coreference Resolution

Step 1: Character mention detection

It is Ella's 11th birthday, and the family is all there to celebrate. Ella looks at her cake waiting for the chance…
Question:
In above story, does Ella refer to people?
LLM output: "yes"

Step 2: Character Coreference Resolution

Annotate all the people mentions within square brackets in the following text with coreference clusters, with the following format [mention](#cluster_name) …
Input:
It is [Ella]'s 11th birthday, and [the family] is all there to celebrate. [Ella] looks at her cake waiting for the chance…

LLM output: It is [Ella](#1)'s 11th birthday, and [the family](#1, #2, #3, #4, #5) is all there to celebrate. [Ella](#1) looks at her cake waiting for the chance…

(b) Textual Coreference Resolution

Figure 3: Illustration of the incremental clustering algorithm for creating visual coreference chains and prompts used in textual coreference resolution. (a) Character detections in Image$_{k+1}$ are compared against character clusters from the previous k images to generate a visual similarity matrix. Best matching character detections are added to existing clusters. Detections that do not match any clusters are grouped into new clusters. (b) A QA-based prompting method is used to identify entities referring to characters (Step 1). Then character clusters are identified using a structured prompt template, which can handle singular and plural character mentions (Step 2).

between multiple overlapping objects—in our case images can contain densely populated background characters. Secondly, we propose to cluster segmentation masks in chains following an incremental algorithm (shown in Figure 3a and Algorithm 1) which does not require the number of clusters to be known in advance and does not depend on facial features, which often fail to detect side-views of faces (in visual stories).

Given an image sequence, we first apply a pretrained object detector and retain detections with the label *Person* and a confidence score higher than 0.9 which cover at least 10% of the image (to remove background characters). Visual chains are constructed incrementally: First, all characters detected in the first image are considered new and added to the chain. Subsequently, for every new character detected in image$_{k+1}$, we compute the pairwise similarity between this character and clusters from previous image$_{0:k}$, thus obtaining a similarity matrix. We construct a bipartite graph and the maximum weighted bipartite matching is computed using the Hungarian algorithm (Kuhn, 1955). Nodes in this bipartite graph represent crops (character instances cropped using their segmentation masks), while edge weights correspond to visual similarities between crops. The algorithm finds a one-to-one alignment that maximizes the sum of the similarities between crops.

We added a threshold condition to the Hungarian algorithm which creates new clusters if new characters are not visually similar to existing ones. Specifically, a new character is added to the visual coreference chains whenever a character detection fails to exceed a predefined similarity threshold. An illustration of our algorithm is shown in Figure 3 and a more formal description is given in Appendix A. We employed DETR (Carion et al., 2020) to detect character bounding boxes, and SAM (Kirillov et al., 2023) to obtain segmentation masks for them. We used OpenCLIP ViT-G/14 (Ilharco et al., 2021) for visual feature extraction and measured the visual similarity between two crops with LLaVA1.5-13B (Liu et al., 2024a) by providing two crops and asking the model whether they represent the same character.

2.1.2 Textual Character Coreference

Current state-of-the-art coreference resolution models generally struggle to effectively handle the coreference of plural nouns (Liu and Keller, 2023; Le and Ritter, 2023; Hicke and Mimno, 2024), which are very common in stories. Our proposed coreference resolution method is based on LLMs (see Figure 3b); through in-context learning, we are able to detect coreference relationships for characters denoting a single *and* multiple entities (i.e., plural and collective nouns). More specifically, our method consists of two steps:

Character Mention Detection. Given a visual story, we detect all entities mentioned therein

using spaCy (Honnibal and Montani, 2023) and mark these with square brackets, denoting a span to annotate in the next step (e.g., James → [James]). We identify character-specific entities with a zero-shot prompting method which takes the story as input and spaCy entities and identifies via question-answering whether these denote persons (either groups or singular entities). Our full prompt is provided in the Appendix (see Table 9) and an illustration in Figure 3b (Step 1).

Character Coreference Resolution. Using in-context learning, we annotate character spans with cluster IDs (e.g., [James] → [James](#1)). We use five examples, and a prompt that instructs the LLM to annotate mentions within square brackets with markdown tags to indicate clusters, resulting in the format [mention](#cluster_name). Note that plural mentions can refer to more than one characters (e.g., [We] → [We](#1, #2)). For an illustration, see Figure 3b (Step 2); the full prompt is given in the Appendix (see Table 11). We used LLAMA3-70B (Touvron et al., 2023) to identify character mentions and obtain textual chains.

2.1.3 Multimodal Character Alignment

Finally, we align textual and visual coreference chains. We again model this alignment as a bipartite graph matching problem. Specifically, we create a matrix where each cell represents the similarity between a textual and visual chain and apply the Hungarian algorithm to find the best match. We measure chain similarity following the method proposed in Liu and Keller (2023) which compares the distribution of characters in images with their distribution in sentences. For instance, if a character is mentioned in the first and second sentence, and there are two coreferent visual detections in the first and second image, then these are likely to refer to the same character.

More formally, a C-dimensional binary vector (where C is the number of images or sentences[2]) represents the distribution of a character. Let C_i denote the i-th textual/visual character, then $C_i[k] = 1$ refers to the i-th character being present in the k-th sentence/image. The similarity between a visual and a textual character is the dot product of their respective binary vectors.

[2]We assume the number of sentences in a story is the same as the number of input images.

Model	Detection			Coreference		
	P	R	F1	P	R	F1
Visual	41.8	78.4	54.53	70.4	74.2	72.25
Textual	84.9	94.5	89.44	77.2	80.3	78.72
Multimodal	—	—	—	39.7	42.5	41.05

Table 1: Results (in %) for character detection and coreference chain identification. P/R are precision/recall.

2.1.4 Quality Evaluation

We evaluated our automatic pipeline against VIST-Character (Liu and Keller, 2023), a high-quality, human-annotated dataset containing rich character annotations for 770 visual stories from the VIST test set (Huang et al., 2016), including visual and textual coreference chains and their alignments. We further split this dataset into 300 stories for validation and 470 stories for testing. Results on the test set are shown in Table 1. We report the precision and recall of automatically detected characters. We assume a textual detection is correct if the head of the noun phrase in question is the same as the gold-standard one. For visual detections, a predicted region is considered correct when the Intersection over Union (IoU) (Yu et al., 2016) is higher than a threshold of 50%. We evaluate coreference chains in an end-to-end setting (i.e., without using gold-standard detections), using the B^3 metrics (Cai and Strube, 2010a), which measure the average percentage of correctly detected mentions in a chain. Specifically, we compute precision and recall for each individual mention (see Appendix D for a formal definition).

In general, we observe good precision and recall for textual character detection and coreference. Visual characters are harder to detect (precision 41.8%), but we are able to build accurate coreference chains for the characters we identify. Multimodal alignment is harder precisely because of visual detection being challenging. Nevertheless we obtain a 10%–15% improvement in multimodal chain accuracy over Liu and Keller (2023) who use simple, unsupervised models for this task based on distributional similarity and CLIP embeddings (Radford et al., 2021). We provide a more detailed comparison in Appendix D.

Finally, in Table 2 we compare VIST++ to existing character-centric visual story generation datasets. VIST-Character (Liu and Keller, 2023)

Dataset	Auto.	Coref.	#Stories	#Chars	#Boxes
VIST-Char	✗	✓	0.7K	3K	5K
VWP	✗	✗	12K	157K	157K
VIST++	✓	✓	**40K**	**300K**	**520K**

Table 2: Statistics for datasets with character annotations. 'Auto' indicates whether the dataset has been annotated automatically. '#stories' and '#chars' refer to the number of visual stories and unique characters.

is manually curated but relatively small-scale, mainly designed for evaluation. Visual Writing Prompts (VWP; Hong et al., 2023) is larger with 12K stories and character annotations provided by humans. However, it does not include coreference chains and consists of images selected from movie scenes. VIST++ contains the highest number of stories and silver-standard annotations; although we focus on VIST, our automatic pipeline could be applied to related datasets such as VWP.

2.2 Visual Story Generation Model

Our generation model is trained on VIST++. It is able to identify characters across multiple images and generate stories with multimodal character chains. We leverage the visual understanding and text generation capabilities of Large Vision-Language models (LVLMs). Specifically, we employ OTTER (Li et al., 2023) as our backbone model, which has been instruction-tuned for multiple image understanding. OTTER (Li et al., 2023) is the instruction-tuned version of OpenFlamingo (Awadalla et al., 2023) (an open-source replication of DeepMind's Flamingo models). Openflamingo comprises a LLaMA-7B (Touvron et al., 2023) language encoder and a CLIP ViT-L/14 (Radford et al., 2021) vision encoder. The two modalities are interleaved through gated cross-attention layers that allow the model to combine information from the visual and textual streams.

Training. We finetune OTTER (Li et al., 2023) on pairs of image sequences and their corresponding stories. In VIST++, characters are outlined with segmentation masks, where same color indicates same characters, i.e, a coreference chain (see Figure 2). Characters are further labeled with a unique ID number, overlaid in the center of the mask, and characters with the same ID in different images are assumed to be the same person. These visual annotations or marks (Yang et al., 2023a) serve as visual prompts to OTTER.

During training, the model learns to predict a story with character grounding and coreference information. Aside from visual prompting, grounding is facilitated by training the model to *verbalize* textual character chains and their grounding. The model learns to predict textual chains, i.e., it marks character mentions with a special symbol, and is explicit about which visual segmentation they refer to (e.g., [Ella](#4) refers to visual segment 4, [she](#4) refers to Ella and segment 4, whereas [We](#1, #2) refers to segments 1 and 2).

Within OTTER, we freeze the parameters of the language encoder (LLAMA-7B) and the vision encoder (CLIP VIT-L/14), and only update the parameters of the Perceiver (Jaegle et al., 2021) resampler module, the cross-attention layers added to the language encoder, and the input/output embeddings of the language encoder. This results in approximately 1.44 billion trainable parameters.

Inference. During inference, the input to the model is a regular image sequence (without any character related annotations). We use the automated pipeline described in Section 2.1.1 to obtain visual character chains which are in turn converted into visual prompts, following the same method used during training. The image sequence with the visual character prompt serves as input to our model which is trained to generate a story with character grounding and coreference information.

3 Experiments

3.1 Datasets

We perform experiments on VIST (Huang et al., 2016), which is the most widely used visual storytelling dataset; it contains 10,117 Flickr albums and 210,819 unique photos. Each training sample consists of $k = 5$ images and a corresponding story of $k = 5$ sentences. Training, validation, and test sets contain 40,155, 4,990, and 5,055 unique stories, respectively. Our model is trained on VIST++, making use of the character-centric annotations discussed in Section 2.1. We report results on the original VIST test set (no annotations) and also on the VIST-Character subset (Liu and Keller, 2023) with gold-standard annotations when evaluating character specific properties.

The Visual Writing Prompts (VWP, Hong et al., 2023) dataset is a benchmark for character-grounded story generation with curated image sequences. However, it is not entirely suited to our task, as it only annotates the bounding box of a character upon their first appearance without providing a coreference chain, preventing us from evaluating the character coreference. Nevertheless, we evaluate our models, trained on the VIST dataset, on VWP to demonstrate their generalization capability, using metrics other than character coreference.

3.2 Implementation Details

We finetuned OTTER with a learning rate of 1e-5, batch size of 32, and warm-up ratio of 0.05. During inference, we employed greedy decoding. We accelerate the training process using the DeepSpeed framework (Rasley et al., 2020).

3.3 Evaluation

Many previous studies (Liu et al., 2023; Hsu et al., 2021a, 2020; Hu et al., 2020; Yang et al., 2019; Modi and Parde, 2019) have highlighted the limitations of metrics based on lexical matching for story evaluation. They correlate poorly with human judgments and do not effectively measure semantic similarity with human-written stories or the lexical richness of the generated stories. In this work, we employ *story-specific* metrics to assess various aspects of story quality, including diversity, grounding, naturalness, and readability. We also propose automatic metrics that assess character richness and the accuracy of coreference chains. Finally, we introduce LLM-as-Judge evaluators (Zheng et al., 2024; Liu et al., 2024b; Liusie et al., 2023) to perform binary comparisons of stories generated by different systems.

Diversity. We use Inter-story Repetition (Yao et al., 2019; Goldfarb-Tarrant et al., 2020), which examines trigram repetition across stories to measure diversity. High inter-story repetition suggests that the model tends to generate the same story even when conditioned on different image sequences.

Grounding. We use GROOViST (Surikuchi et al., 2023) to assess whether the stories accurately represent the content of the image sequences. GROOViST employs CLIPScore (Hessel et al., 2021) to compute the similarity between noun phrases in the story and bounding boxes in the images, producing an average score which favors stories with concrete words as they are more likely to be visible.

Naturalness. We use MAUVE (Pillutla et al., 2021) to measure the naturalness of the stories. MAUVE has a high correlation with human judgments and computes the similarity between the distributions of human- and machine-generated texts.

Reading Difficulty. We measure the reading difficulty of stories using the Flesch-Kincaid Grade Level (FKGL; Kincaid et al. 1975) formula:

$$\text{FKGL} = a\frac{N_{word}}{N_{sentence}} + b\frac{N_{syllable}}{N_{word}} + c \quad (1)$$

where a is 0.39, b is 11.8, and c is -15.59 as defined in Kincaid et al. (1975). The FKGL values denote reading age, ranging from 0 to 18, So, lower values suggest the text is easier to read and higher values denote increased reading difficulty.

Character Richness and Coreference. We report the number of characters and character mentions in generated stories as measures of character richness. We also evaluate the accuracy of multimodal coreference chains with B^3 precision (Cai and Strube, 2010b), as we are more interested in the correctness of mentions within a predicted chain rather than ensuring that every gold-standard visual character is mentioned. For comparison systems which do not detect characters or ground them to images, we employ the pipeline described in Section 2.1.2 on their generated output. Note that character metrics are computed against the VIST-Character (Liu and Keller, 2023) test set, as it contains manually annotated visual coreference chains.

LLM as a Judge. We further introduce an LLM-based evaluator to perform side-by-side comparisons of system output. Specifically, we employed human side-by-side judgments of visual stories from previous work (Hsu et al., 2022; Liu et al., 2023) covering the following criteria: Specificity, Coherence, Engagement, Grounding, Characters, and Overall Preference. We then designed a prompt targeting each dimension of story quality

Input	SPE	COH	ENG	GRD	CHA	OVR
Multimodal	81.24	75.26	74.23	74.52	70.52	75.26
Text-Only	78.10	71.36	75.19	—	65.11	74.43

Table 3: Accuracy of GPT-4o against human pair-wise judgments of story quality. SPE, COH, ENG, GRD, CHA, OVR are short for specificity, coherence, engagement, grounding, characters, and overall preference.

(see Table 12 in the Appendix) and evaluated LLM accuracy against human judgments.

Table 3 reports the accuracy of GPT-4o as a judge (more details in Appendix B). We present results for LLMs like GPT-4o, which can process images *and* text, but also simulate the use of text-only LLMs by giving GPT-4o the story without the associated image sequence. GPT-4o agrees with human judgments 70–80% of the time across the story quality dimensions. This suggests that it is a fairly accurate evaluator and could replace costly and time-consuming human judgments.

The d_{HM} Evaluation Metric. A concurrent study by Surikuchi et al. (2024) introduces a method that assesses the quality of generated stories in terms of their distance from human-written stories. The proposed d_{HM} metric (a shorthand for ''distance between humans and models'') combines three reference-free evaluation metrics: GROOViST (Surikuchi et al., 2023) for visual grounding, RoViST-C for coherence, and RoViST-NR for non-redundancy/repetition. We incorporate these metrics into our analysis to ensure a more comprehensive and up-to-date evaluation.

4 Results

Our experimental results are summarized in Table 4. The first rows present the performance of state-of-the-art storytelling systems from the literature. **MCSM** (Chen et al., 2021) exploits a commonsense knowledge graph to represent concepts depicted in the images and their relations, and uses a **M**aximal **C**lique **S**election **M**odule to identify which ones to write a story about. MCSM uses BART-large (Lewis et al., 2020) to generate the story based on selected concepts (and image features). **Iter-Blueprint** (Liu et al., 2023) is built on top of BART-base and leverages a sequence of question-answer pairs as a blueprint (story plan) for selecting salient visual concepts and determining how to present them in the story. The model employs an incremental generation strategy, gradually constructing the blueprint and its corresponding story sentence-by-sentence.

SOTTER++ is our story generation model based on OTTER and finetuned on the proposed VIST++ dataset, whereas **SOTTER** is finetuned on vanilla VIST and does not have any specific knowledge about characters or their grounding. Finally, Table 4 includes results for **GPT-4V**[3] (zero-shot setting, prompt in Figure 10) which is one of the most performant multimodal LLMs. All models in Table 4 were evaluated on the same VIST test set.

SOTTER++ Leads on Character-centric Metrics. The number of characters and their mentions in stories generated by SOTTER is similar to that of previous visual storytelling systems. However, the number of characters and frequency of character mentions increase substantially for SOTTER++, which is trained on VIST++. In addition, the accuracy of the coreference chains improves by more than 5%. This indicates that VIST++ effectively enables LVLMs to generate stories with richer characters (more unique characters and mentions) and more accurate character chains (higher coreference scores). Although GPT-4V generates more characters than SOTTER++, it does so at the expense of succinctness and readability, whereas SOTTER++ produces stories with lengths and readability scores closer to those of human-written stories.

LLM-based Models Improve Story Diversity. We observe that LLM-based models (i.e., OTTER variants, ∼9B parameters; GPT-4V, undisclosed parameter count) generate more diverse stories compared to MCSM and Iter-Blueprint, which are based on smaller language models like BART (∼300M parameters). This indicates that the extensive prior knowledge of LLMs prevent them from overfitting on the VIST dataset, which would otherwise result in similar stories for different images. Additionally, SOTTER++ creates more diverse stories than SOTTER. Character coreference annotations help the model generate stories that are more image specific, avoiding repetition. Table 5 further demonstrates that SOTTER++ generates

[3] https://openai.com/index/gpt-4v-system-card/.

Model	Character Metrics (↑)			Story Metrics					N-gram-based Metrics (↑)			
	#CHAR	#MENTS	COREF	#Words	RDD	DIV (↓)	GRD (↑)	MAU (↑)	B-4	RL	MET	CID
MCSM	1.21	2.01	10.43	39.11	5.30	77.48	48.12	11.01	8.1	27.7	31.4	7.6
Iter-Blueprint	1.52	2.13	12.56	38.67	6.10	72.70	62.54	**28.25**	7.0	26.1	30.3	5.5
SOTTER	1.91	3.15	26.61	50.18	5.12	56.10	**65.97**	10.68	5.4	18.2	23.1	2.9
SOTTER++	**2.98**	**4.98**	**32.16**	51.75	5.42	**54.58**	64.70	10.47	5.1	18.0	22.6	2.6
GPT-4V (0-shot)	5.99	9.76	29.96	118.34	12.86	9.61	58.30	2.41	0.8	15.2	15.7	0.1
GPT-4V (<50 wds.)	1.98	3.99	30.51	56.24	9.89	8.00	61.68	2.40	1.7	14.4	16.6	5.0
GPT-4V (5-shot)	2.55	4.23	29.98	74.57	10.95	15.30	48.15	5.18	1.5	13.5	16.5	0.6
GPT-4V (5-shot++)	2.67	4.99	28.22	86.15	11.86	11.68	49.28	4.93	1.5	14.0	16.3	0.6
Gold	4.11	5.09	51.36	48.95	5.51	37.24	73.60	—	—	—	—	—

Table 4: Automatic evaluation results. We report character-centric metrics, story-specific metrics, and metrics measuring lexical similarity between system stories and their references. GPT-4V (5-shot++) shows the performance of GPT-4V when provided with five visual stories with gold-standard multimodal character annotations and co-reference chains. The abbreviations #CHAR, #MENTS, COREF correspond to the number of characters, mentions, and the coreference score. DIV, GRD, MAU, RDD, B-4, RL, MET, and CID are short for Diversity (lower is better), Grounding, MAUVE, Reading Difficulty, Blue-4, RLSum, METEOR, and CIDER, respectively. Best results, excluding GPT-4V, are highlighted in bold.

Model	GROOViST	RoViST-C	RoViST-NR
MCSM	48.12	0.67	0.90
Inter-Blueprint	62.54	0.72	0.93
SOTTER	65.97	0.78	0.95
SOTTER++	64.70	0.80	0.95

Table 5: Evaluation results based on the human-centric metrics proposed by Surikuchi et al. (2024). Higher scores indicate better performance in visual grounding (GROOViST), coherence (RoViST-C), and non-redundancy (RoViST-NR).

stories that are more grounded and more diverse compared to previous work.

Existing Metrics Fall Short of Evaluating Character Grounding. SOTTER++ performs best in character-centric metrics but is slightly worse than SOTTER in terms of grounding (see column GRD in Table 4). Our analysis suggests that this discrepancy arises because existing grounding metrics, such as GROOViST (Surikuchi et al., 2023) rely on general-purpose entity extraction and use CLIPScore to evaluate grounding between text and images. As a result, they do not accurately measure the grounding of character mentions, particularly when it comes to character names and pronouns (Liu and Keller, 2023).

MAUVE and n-gram Metrics Understimate the Performance of LLM-based Models. As shown in Table 4, LLM-based models (most dramatically GPT-4V) have lower MAUVE scores than models trained from scratch on VIST. This is because MAUVE compares the learned distribution from a story generation model to the distribution of human-written stories. Consequently, the prior knowledge acquired during LLM pre-training may lead to less accurate alignment with the distribution of human-authored stories, resulting in lower MAUVE. We also observe in Table 4 that MAUVE and Diversity exhibit an inverse relationship, suggesting that human stories in VIST may themselves lack diversity, and we would expect models more aligned with these stories to have lower diversity.

We observe an analogous pattern for n-gram metrics (Table 4, right block), including Blue-4, RLSum, METEOR, and CIDER. These metrics underestimate the performance of LLM-based models compared to models trained from scratch (MCSM, Iter-Blueprint). This is because they measure how well a model mimics the human stories in terms of word-overlap, rather than in terms of diversity, grounding, or character-centricity.

GPT-4V Tends to Generate Rare Words and Complicated Stories. Reading difficulty scores suggest that the stories generated by GPT-4V are more complex, with longer sentences and rare words. Even with five demonstrations for in-context learning, GPT-4V still cannot match the reading difficulty of the VIST stories. When

VCoref	TCoref	#CHAR	#MENTS	COREF
✗	✗	1.91	3.15	26.61
✓	✗	1.89	3.19	27.54
✗	✓	2.44	3.56	27.14
✓	✓	2.98	4.98	32.16

Table 6: Ablation study assessing the contribution of different annotation components. VCoref stands for visual character coreference and TCoref, for textual character coreference.

explicitly instructed in the prompt to generate stories with fewer than 50 words, it produces stories that are on average closer to this length, but reading difficulty remains higher than the gold standard and other models. In contrast, fine-tuned models better match the readability of stories written by humans. When GPT-4V is given five visual stories with gold-standard multimodal character annotations and co-reference chains (5-shot++ in Table 4), the most notable change is a further increase in story length compared to regular 5-shot prompting. We attribute this to the added complexity of the demonstrations, which makes in-context learning more challenging rather than improving character quality. This is supported by a 1.7-point drop in the COREF score. Additionally, the slight increase in the number of characters and mentions comes at the cost of longer stories. These findings suggest that in-context learning is not effective in character-centric visual storytelling tasks.

Combining Visual and Textual Coreference Annotations Improves Character Coreference. The ablation in Table 6 assesses the impact of visual and textual character coreference. When we only annotate characters in the input images, we do not observe an improvement in the number of characters, their mentions, or the coreference in the generated stories (compare the first and second row in Table 6). This shows that despite visually indicating which characters are the same in the images, the output stories remain practically unchanged and the model is unable to utilize visual character annotations effectively.

When the model is trained on stories annotated with textual coreference chains, the predicted stories feature richer character representations and more mentions (see third row in Table 6). This indicates that applying character annotations to the target data is a more direct and effective approach,

	vs Iter-BP			vs SOTTER			vs Human		
Criteria	W	L	T	W	L	T	W	L	T
SPE	**82.3**	9.5	8.2	**50.9**	44.0	5.2	41.0	**53.2**	5.8
COH	**81.5**	13.1	5.5	41.6	**47.8**	10.6	42.1	**51.0**	6.9
ENG	**64.9**	26.1	9.1	**42.3**	42.2	15.4	36.0	**56.2**	7.7
GRD	**71.8**	20.6	7.6	**57.4**	40.3	2.3	37.9	**56.0**	6.0
CHA	**88.4**	8.0	3.7	**75.1**	18.2	6.7	**51.6**	46.8	1.6
OVR	**75.0**	22.0	3.0	**70.1**	22.1	7.8	36.9	**55.0**	8.1

Table 7: Pair-wise comparison evaluation using GPT-4o as a judge. We report the percentage of times SOTTER++ Wins (W), Loses (L), or Ties (T) with a comparison system. We evaluate story specificity (SPE), coherence (COH), engagement (ENG), grounding (GRD), characters (CHA), and overall (OVR).

allowing the model to focus more on characters during generation. However, we also observe that, in this setting, coreference accuracy does not improve significantly, indicating that the model is not better at identifying which characters in the images are the same person.

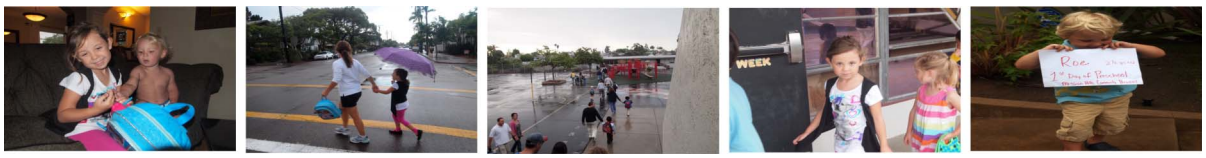
When combining visual and textual coreference annotations (last row in Table 6), coreference scores increase (and the number of characters and mentions). This indicates that character annotations in both modalities are complementary, enabling the model to recognize which characters are the same, leading to more accurate coreference.

SOTTER++ Leads in Side-by-side Evaluation. Table 7 summarizes pair-wise comparison results using GPT-4o as an evaluator. Specifically, we compare SOTTER++ against (a) the iterative Blueprint model (Iter-BP); (b) SOTTER trained on VIST without character annotations; and (c) gold-standard stories written by humans. We observe that SOTTER++ outperforms the iterative Blueprint model across all metrics and is superior to SOTTER across most dimensions (with the exception of coherence). When compared to gold-standard stories, SOTTER++ achieves a win rate of 36.9%, and is better at generating stories with explicit and recurring characters (see characters metric). This suggests that even human-written stories in the test set tend to use vague plural pronouns and avoid clear and recurring character usage.

SOTTER++ Generalizes to the VWP Dataset. Table 8 presents the results of evaluating models on the VWP dataset, when trained on VIST

Model	Character Metrics (↑)			Story Metrics					N-gram-based Metrics (↑)			
	#CHAR	#MENTS	COREF	#Words	DIV (↓)	GRD (↑)	MAU (↑)	RDD	B-4	RL	MET	CID
MCSM	—	—	—	—	—	—	—	—	—	—	—	—
Iter-Blueprint	1.66	2.55	—	57.34	71.11	54.34	**22.25**	8.52	6.1	25.1	31.1	5.2
SOTTER	2.01	3.22	—	51.54	57.12	59.47	9.06	8.01	5.6	18.6	23.0	2.7
SOTTER++	**2.97**	**4.88**	—	52.04	**53.48**	**61.21**	10.87	8.16	5.7	18.9	22.6	2.6

Table 8: Out-of-domain generalization results on the VWP dataset. We were unable to evaluate against MCSM, since their code is not available. Iter-Blueprint and SOTTER were trained on VIST, and SOTTER++ on VIST++.



MCSM: Today was [female]'s first day of school. She made her first snowboard run on [female]'s first day. There was a lot of people at the race. When the race started, [female] and her mom helped the kids outside. [female]'s baby was born on her first day of school and her mom helped [female] with her feet.

Iter-BP: The oldest daughter is being present for a birthday party. She and her friends decided to have a walk throughout the neighbourhood. Some of her friends came to at the exhibition. The children smiled and had a fun time smiling for THE camera!

SOTTER: We got ready for the first day of school today. It was raining really hard so we had to bring the umbrella. All the children made their way into the building. Here is our daughter getting ready for her first classes. Our son was really excited to go.

SOTTER++: Today was the first day of school for [my little girl](1). [She](1) was so excited to see [her friends](2) and [teachers](3) again. [We](1, 4) had a great time walking to the school together. [The kids](1, 2) were so happy to see each other. [I](4) am so proud of [her](1)! After the school day is over, [her brother](5) is waiting for [her](1) and hear all about her day.

GPT-4V: Roe and her younger brother prepared excitedly for her first day of preschool, packing her favorite blue backpack. Despite the rain, their mother walked them to school, sharing an umbrella and keeping their spirits high. As they approached the school gates, Roe saw other children and parents bustling about, some hurrying to avoid the downpour. Inside, she bravely entered the classroom, glancing back to reassure her little brother, who watched with wide eyes. At the end of the day, Roe proudly displayed her "1st Day of Preschool" sign, marking a successful start to her educational journey.

Human: She doesn't want to share her candy with her little brother before she goes to school. Sally walking Dora to school in the rain. All the kids showing up for school. Hope it stops raining so the kids can go outside today. Dora didn't want another picture, but she's just so cute. He's waiting for Dora to get home from school. He always misses her.

Figure 4: Examples of system output and human-written story for an image sequence (VIST test set).

and VIST++. Since VWP contains more images per story than VIST, the average story length is correspondingly longer. As can be seen, SOTTER++ generalizes best on VWP, outperforming comparison models in character-related metrics, grounding, and diversity.

SOTTER++ Generates Character-centric Swith Consistent Mentions. Figure 4 shows example stories created by the models used in our evaluations, as well as by GPT-4V. We observe that previous state-of-the-art models, such as MCSM, have poor character mention capabilities, replacing character names with [female] or [male]. While the stories generated by Iter-BP exhibit more human-like language, they contain hallucinations, as evidenced by the low Grounding scores in Table 4. Additionally, most characters in the story are generic (e.g., friends), lacking explicit mentions to recurring individuals.

In the stories produced by SOTTER++, we observe recurring characters represented by consistent coreference cluster labels. Even for plural characters (denoted by ‘‘they’’, for instance), SOTTER++ predicts the specific individuals they refer to. Furthermore, the story language is similar to the style of the human story. GPT-4V stories have the highest reading difficulty (RDD; see Table 4): They are long, use rare words, and employ a flamboyant style which differs from how humans write visual stories. In contrast, fine-tuned models better match the readability level of human-authored stories. There is also a significant error in the story generated by GPT-4V in Figure 4: It incorrectly assumes that Roe is holding a sign (last story sentence), whereas the image depicts

her younger brother. This suggests that even powerful LVLMs like GPT-4V may struggle to maintain character identity across a sequence of images, either due to limitations in recognizing the same character or because they do not consistently attend to this information as a meaningful feature.

5 Related Work

Visual Storytelling. Huang et al. (2016) proposed visual storytelling as a way to create AI systems that can comprehend event sequences and generate narrative language which goes beyond just describing what is shown in images. Early methods (Gonzalez-Rico and Fuentes-Pineda, 2018; Kim et al., 2018b) used simple encoder-decoder models with CNNs for visual features and RNNs for text generation. Recent approaches (Wang et al., 2024; Chen et al., 2021; Hsu et al., 2020; Yang et al., 2019) leverage external knowledge resources like ConceptNet to enhance commonsense reasoning abilities. Some methods (Lu et al., 2016; Hong et al., 2020; Wang et al., 2020) also utilize scene graphs to model object relationships.

Few approaches have focused on character-centric story generation. Surikuchi and Laaksonen (2019) extract characters from VIST, analyze their relationships and exploit them for focusing attention to relevant visual segments during story generation. However, neither character coreference nor character grounding are explicitly modeled. Liu and Keller (2023) introduce the VIST-Character dataset with rich character and coreference annotations, however due to its small size, it can only serve as a test set.

Most existing approaches (Xu et al., 2021; Hsu et al., 2020; Wang et al., 2020; Yang et al., 2019) train Transformer models from scratch, with the exception of Liu et al. (2023) and Chen et al. (2021), who employ a vanilla BART model as a baseline without task-specific adaptation. In contrast, our work leverages the language modeling and generalization capabilities of LLMs (i.e., larger than 1B parameters) aiming to generate visually grounded and character-centric stories.

Character Coreference in Visual Stories. Previous visual coreference methods (Schroff et al., 2015; Liu and Keller, 2023) rely on clustering algorithms which mostly depend on facial features and require a predefined number of clusters. Facial features often fail to detect side-view faces, which are common in visual stories, and since deciding the number of clusters is non-trivial, existing methods often identify different characters in an image as the same person. Our approach eschews these issues by focusing on segmentation masks and adopting an incremental algorithm which does not require the number clusters to be known in advance. Hong et al. (2023) create Visual Writing Prompts (VWP), a dataset which contains visual stories based on movie shot sequences and manual annotations of characters aligned to images but no coreference chains. They also demonstrate that character prompts lead to more diverse and visually grounded stories. However, in their approach grounding is implicit, whereas we force the model to align textual mentions to visual detections via markdown tags.

Current state-of-the-art textual coreference resolution models generally struggle to effectively handle the coreference of plural nouns (Liu and Keller, 2023; Le and Ritter, 2023; Hicke and Mimno, 2024), which are very common in visual storytelling. Our prompting approach based on LLMs can effectively handle plural and collective nouns and resolve them to unique characters, overcoming the limitations of existing methods.

Characters in Textual Story Generation. Modeling characters has been shown to significantly improve story quality by enabling systems to reason about inter-character relationships (Vijjini et al., 2022; Brahman and Chaturvedi, 2020; Clark et al., 2018). Beyond generation, understanding characters is also crucial for narrative comprehension. Prior work has explored character identification (Vala et al., 2015), dynamic relationship modeling (Chaturvedi et al., 2017; Iyyer et al., 2016), and emotional characterization (Kim and Klinger, 2019), highlighting the central role of characters in storytelling tasks. Our approach builds on these insights by explicitly modeling character grounding and coreference in visual stories.

6 Conclusions

This work introduces a character-centric approach to visual story generation. We first propose VIST++, a new dataset that extends VIST with visual and textual character grounding and coreference annotations. The dataset comprises

annotations for 300K unique characters over 40K visual stories. We then present a new model for character-centric story generation, built on the pre-trained OTTER LVLM and fine-tuned on VIST++. Our evaluation shows that the model generates stories with rich characters and accurate coreference. Furthermore, using a novel LLM-as-a-Judge evaluation, we demonstrate that stories generated by our model are preferred over those without character grounding.

Limitations Several visual storytelling datasets have recently emerged (Ravi et al., 2021; Hong et al., 2023). However, our work focuses on VIST due to its continued popularity and to minimize the substantial costs associated with new annotation and experimentation. While our character-centric method improves story generation, an important avenue for future work lies in improving coreference scores, which would likely lead to even greater story quality.

Acknowledgments

The authors gratefully acknowledge the support of the UK Engineering and Physical Sciences Research Council (grant EP/W002876/1). Liu was supported by the UKRI Centre for Doctoral Training in Natural Language Processing, funded by the UKRI (grant EP/S022481/1) and the University of Edinburgh.

References

Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. 2023. OpenFlamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*.

David Bamman, Brendan O'Connor, and Noah A. Smith. 2013. Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361, Sofia, Bulgaria. Association for Computational Linguistics.

Faeze Brahman and Snigdha Chaturvedi. 2020. Modeling protagonist emotions for emotion-aware storytelling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5277–5294, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.426

Faeze Brahman, Meng Huang, Oyvind Tafjord, Chao Zhao, Mrinmaya Sachan, and Snigdha Chaturvedi. 2021. ''Let your characters tell their story'': A dataset for character-centric narrative understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1734–1752, Punta Cana, Dominican Republic. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.findings-emnlp.150

Jie Cai and Michael Strube. 2010a. Evaluation metrics for end-to-end coreference resolution systems. In *Proceedings of the SIGDIAL 2010 Conference*, pages 28–36.

Jie Cai and Michael Strube. 2010b. Evaluation metrics for end-to-end coreference resolution systems. In *Proceedings of the SIGDIAL 2010 Conference*, pages 28–36. Tokyo, Japan. Association for Computational Linguistics.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer. https://doi.org/10.1007/978-3-030-58452-8_13

Snigdha Chaturvedi, Mohit Iyyer, and Hal Daume III. 2017. Unsupervised learning of evolving relationships between literary characters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31. https://doi.org/10.1609/aaai.v31i1.10982

Hong Chen, Yifei Huang, Hiroya Takamura, and Hideki Nakayama. 2021. Commonsense knowledge aware concept selection for diverse and informative visual storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 999–1008. https://doi.org/10.1609/aaai.v35i2.16184

Jiangjie Chen, Xintao Wang, Rui Xu, Siyu Yuan, Yikai Zhang, Wei Shi, Jian Xie, Shuang Li, Ruihan Yang, Tinghui Zhu, Aili Chen, Nianqi Li, Lida Chen, Caiyu Hu, Siye Wu, Scott Ren, Ziquan Fu, and Yanghua Xiao. 2024. From persona to personalization: A survey on role-playing language agents. Version Number: 1.

Nuo Chen, Yan Wang, Haiyun Jiang, Deng Cai, Yuhan Li, Ziyang Chen, Longyue Wang, and Jia Li. 2023. Large language models meet harry potter: A dataset for aligning dialogue agents with characters. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8506–8520. https://doi.org/10.18653/v1/2023.findings-emnlp.570

Elizabeth Clark, Yangfeng Ji, and Noah A. Smith. 2018. Neural text generation in stories using entity representations as context. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2250–2260. https://doi.org/10.18653/v1/N18-1204

Seraphina Goldfarb-Tarrant, Tuhin Chakrabarty, Ralph Weischedel, and Nanyun Peng. 2020. Content planning for neural story generation with aristotelian rescoring. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4319–4338, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.351

Diana Gonzalez-Rico and Gibran Fuentes-Pineda. 2018. Contextualize, show and tell: A neural visual storyteller. *arXiv preprint arXiv:1806.00738*.

Alexander Gurung and Mirella Lapata. 2024. CHIRON: Rich character representations in long-form narratives. *arXiv preprint arXiv:2406.10190*.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*.

Rebecca M. M. Hicke and David Mimno. 2024. [lions: 1] and [tigers: 2] and [bears: 3], oh my! Literary coreference annotation with LLMs. *arXiv preprint arXiv:2401.17922*.

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. https://doi.org/10.1162/tacl_a_00553

Xudong Hong, Rakshith Shetty, Asad Sayeed, Khushboo Mehra, Vera Demberg, and Bernt Schiele. 2020. Diverse and relevant visual storytelling with scene graph embeddings. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 420–430. https://doi.org/10.18653/v1/2020.conll-1.34

Matthew Honnibal and Ines Montani. 2023. spacy: Industrial-strength natural language processing in python. Version 3.x.

Chao-Chun Hsu, Zi-Yuan Chen, Chi-Yang Hsu, Chih-Chia Li, Tzu-Yuan Lin, Ting-Hao Huang, and Lun-Wei Ku. 2020. Knowledge-enriched visual storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7952–7960. https://doi.org/10.1609/aaai.v34i05.6303

Chi-Yang Hsu, Yun-Wei Chu, Vincent Chen, Kuan-Chieh Lo, Chacha Chen, Ting-Hao Huang, and Lun-Wei Ku. 2022. Learning to rank visual stories from human ranking data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6365–6378, Dublin, Ireland. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.441

Chi-yang Hsu, Yun-Wei Chu, Ting-Hao Huang, and Lun-Wei Ku. 2021a. Plot and rework: Modeling storylines for visual storytelling. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4443–4453. https://doi.org/10.18653/v1/2021.findings-acl.390

Chi-Yang Hsu, Yun-Wei Chu, Tsai-Lun Yang, Ting-Hao Huang, and Lun-Wei Ku. 2021b. Stretch-VST: Getting flexible with visual stories. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 356–362.

https://doi.org/10.18653/v1/2021.acl-demo.42

Junjie Hu, Yu Cheng, Zhe Gan, Jingjing Liu, Jianfeng Gao, and Graham Neubig. 2020. What makes a good story? Designing composite rewards for visual storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7969–7976. https://doi.org/10.1609/aaai.v34i05.6305

Ting-Hao 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. https://doi.org/10.18653/v1/N16-1147

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. 2021. Openclip. If you use this software, please cite it as below.

Naoya Inoue, Charuta Pethe, Allen Kim, and Steven Skiena. 2022. Learning and evaluating character representations in novels. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1008–1019. https://doi.org/10.18653/v1/2022.findings-acl.81

Mohit Iyyer, Anupam Guha, Snigdha Chaturvedi, Jordan Boyd-Graber, and Hal Daumé III. 2016. Feuding families and former friends: Unsupervised learning for dynamic fictional relationships. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1534–1544. https://doi.org/10.18653/v1/N16-1180

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. 2021. Perceiver: General perception with iterative attention. In *International Conference on Machine Learning*, pages 4651–4664. PMLR.

Evgeny Kim and Roman Klinger. 2019. Frowning frodo, wincing leia, and a seriously great friendship: Learning to classify emotional relationships of fictional characters. *arXiv preprint arXiv:1903.12453*.

Hannah Kim, Denys Katerenchuk, Daniel Billet, Haesun Park, and Boyang Albert Li. 2018a. Learning joint Gaussian representations for movies, actors, and literary characters. *ArXiv*.

Taehyeong Kim, Min-Oh Heo, Seonil Son, Kyoung-Wha Park, and Byoung-Tak Zhang. 2018b. GLAC net: Glocal attention cascading networks for multi-image cued story generation. *arXiv preprint arXiv:1805.10973*.

J. Peter Kincaid, Robert P. Fishburne Jr, Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. https://doi.org/10.21236/ADA006655

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026. https://doi.org/10.1109/ICCV51070.2023.00371

Harold W. Kuhn. 1955. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97. https://doi.org/10.1002/nav.3800020109

Nghia T. Le and Alan Ritter. 2023. Are large language models robust zero-shot coreference resolvers? *arXiv preprint arXiv:2305.14489*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880. https://doi.org/10.18653/v1/2020.acl-main.703

Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Fanyi Pu, Jingkang Yang, Chunyuan Li, and Ziwei Liu. 2023. MIMIC-IT: Multi-modal in-context instruction tuning. *arXiv preprint arXiv:2306.05425*.

Yingxuan Li, Ryota Hinami, Kiyoharu Aizawa, and Yusuke Matsui. 2024. Zero-shot character identification and speaker prediction in comics via iterative multimodal fusion. Version Number: 2.

Danyang Liu and Frank Keller. 2023. Detecting and grounding important characters in visual stories. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13210–13218. https://doi.org/10.1609/aaai.v37i11.26551

Danyang Liu, Mirella Lapata, and Frank Keller. 2023. Visual storytelling with question-answer plans. *arXiv preprint arXiv:2310.05295*.

Danyang Liu, Juntao Li, Meng-Hsuan Yu, Ziming Huang, Gongshen Liu, Dongyan Zhao, and Rui Yan. 2020. A character-centric neural model for automated story generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 1725–1732. https://doi.org/10.1609/aaai.v34i02.5536

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024a. LLaVa-neXT: Improved reasoning, OCR, and world knowledge.

Yinhong Liu, Han Zhou, Zhijiang Guo, Ehsan Shareghi, Ivan Vulic, Anna Korhonen, and Nigel Collier. 2024b. Aligning with human judgement: The role of pairwise preference in large language model evaluators. *arXiv preprint arXiv:2403.16950*.

Adian Liusie, Potsawee Manakul, and Mark J. F. Gales. 2023. Zero-shot NLG evaluation through pairware comparisons with LLMs. *arXiv preprint arXiv:2307.07889*.

Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. 2016. Visual relationship detection with language priors. In *European Conference on Computer Vision*, pages 852–869. Springer. https://doi.org/10.1007/978-3-319-46448-0_51

Yatri Modi and Natalie Parde. 2019. The steep road to happily ever after: An analysis of current visual storytelling models. In *Proceedings of the Second Workshop on Shortcomings in Vision and Language*, pages 47–57. https://doi.org/10.18653/v1/W19-1805

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. MAUVE: Measuring the gap between neural text and human text using divergence frontiers. In *NeurIPS*.

Vadim A. Porvatov, Carlo Strapparava, and Marina Tiuleneva. 2024. Big-five backstage: A dramatic dataset for characters personality traits & gender analysis. In *Proceedings of the Workshop on Cognitive Aspects of the Lexicon @ LREC-COLING 2024*, pages 114–119, Torino, Italia. ELRA and ICCL. https://doi.org/10.63317/3dgdgn6675fa

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506. https://doi.org/10.1145/3394486.3406703

Hareesh Ravi, Kushal Kafle, Scott Cohen, Jonathan Brandt, and Mubbasir Kapadia. 2021. AESOP: Abstract encoding of stories, objects, and pictures. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2052–2063. https://doi.org/10.1109/ICCV48922.2021.00206

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823. https://doi.org/10.1109/CVPR.2015.7298682

Aditya Surikuchi and Jorma Laaksonen. 2019. Character-centric storytelling. *arXiv preprint arXiv:1909.07863*.

Aditya K. Surikuchi, Raquel Fernández, and Sandro Pezzelle. 2024. Not (yet) the whole story: Evaluating visual storytelling requires more than measuring coherence, grounding, and repetition. *arXiv preprint arXiv:2407.04559*.

Aditya K. Surikuchi, Sandro Pezzelle, and Raquel Fernández. 2023. Groovist: A metric for grounding objects in visual storytelling. *arXiv preprint arXiv:2310.17770*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Hardik Vala, David Jurgens, Andrew Piper, and Derek Ruths. 2015. Mr. bennet, his coachman, and the archbishop walk into a bar but only one of them gets recognized: On the difficulty of detecting characters in literary texts. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 769–774. https://doi.org/10.18653/v1/D15-1088

Anvesh Rao Vijjini, Faeze Brahman, and Snigdha Chaturvedi. 2022. Towards inter-character relationship-driven story generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8970–8987, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.emnlp-main.613

Eileen Wang, Caren Han, and Josiah Poon. 2024. SCO-VIST: Social interaction commonsense knowledge-based visual storytelling. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1602–1616, St. Julian's, Malta. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.eacl-long.96

Ruize Wang, Zhongyu Wei, Piji Li, Qi Zhang, and Xuanjing Huang. 2020. Storytelling from an image stream using scene graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9185–9192. https://doi.org/10.1609/aaai.v34i05.6455

Chunpu Xu, Min Yang, Chengming Li, Ying Shen, Xiang Ao, and Ruifeng Xu. 2021. Imagine, reason and write: Visual storytelling with graph knowledge and relational reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3022–3029. https://doi.org/10.1609/aaai.v35i4.16410

Rui Xu, Xintao Wang, Jiangjie Chen, Siyu Yuan, Xinfeng Yuan, Jiaqing Liang, Zulong Chen, Xiaoqing Dong, and Yanghua Xiao. 2024. Character is destiny: Can large language models simulate persona-driven decisions in role-playing? Version Number: 1. https://doi.org/10.48550/arXiv.2404.12138

Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. 2023a. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*.

Kevin Yang, Dan Klein, Nanyun Peng, and Yuandong Tian. 2023b. DOC: Improving long story coherence with detailed outline control. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3378–3465, Toronto, Canada. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.190

Kevin Yang, Yuandong Tian, Nanyun Peng, and Dan Klein. 2022. In *Re3: Generating longer stories with recursive reprompting and revision*. Version Number: 3. https://doi.org/10.18653/v1/2022.emnlp-main.296

Lingfeng Yang, Yueze Wang, Xiang Li, Xinlong Wang, and Jian Yang. 2024. Fine-grained visual prompting. *Advances in Neural Information Processing Systems*, 36. https://doi.org/10.52202/075280-1086

Pengcheng Yang, Fuli Luo, Peng Chen, Lei Li, Zhiyi Yin, Xiaodong He, and Xu Sun. 2019. Knowledgeable storyteller: A commonsense-driven generative model for visual storytelling. In *IJCAI*, pages 5356–5362. https://doi.org/10.24963/ijcai.2019/744

Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7378–7385. https://doi.org/10.1609/aaai.v33i01.33017378

Jiahui Yu, Yuning Jiang, Zhangyang Wang, Zhimin Cao, and Thomas Huang. 2016. UnitBox: An advanced object detection network. In *Proceedings of the 24th ACM International Conference on Multimedia*, pages 516–520. https://doi.org/10.1145/2964284.2967274

Mo Yu, Qiujing Wang, Shunchi Zhang, Yisi Sang, Kangsheng Pu, Zekai Wei, Han Wang, Liyan Xu, Jing Li, Yue Yu, and Jie Zhou. 2022. Few-shot character understanding in movies as an assessment to meta-learning of theory-of-mind. Version Number: 2. https://doi.org/10.48550/arXiv.2211.04684

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. 2024. Judging LLM-as-a-judge with MT-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

Algorithm 1 Incremental Character Matching

Input: Image sequence $\{I_i\}_{i=1}^{K}$
Output: Visual coreference chains $\{\mathcal{V}_i\}_{i=1}^{N_v}$ **Start:**

Initialize visual coref chains $\{\mathcal{V}_i\}_{i=1}^{N_v} \leftarrow \emptyset$
Detect characters in I_1 using DETR, obtain detections D_1
for each detection $d \in D_1$ **do**
    Add d to a new chain in $\{\mathcal{V}_i\}$
end for
for $i = 2$ to K **do**
    Detect characters in I_i using DETR, obtain detections D_i
    Extract visual features of D_i using CLIP
    Initialize similarity matrix $M_{q \times p}$, where $q = |D_i|$ and $p = |\{\mathcal{V}_i\}|$
    for each detection $d_q \in D_i$ **do**
        for each existing character $c_p \in \{\mathcal{V}_i\}$ **do**
            Compute matching score $M[q, p]$ between d_q and c_p using CLIP features
        end for
    end for
    Solve Bipartite Graph Matching problem using Hungarian algorithm on M
    for each detection $d_q \in D_i$ **do**
        if d_q matches to existing character c_p **then**
            Add d_q to the chain of c_p
        else
            Create a new chain in $\{\mathcal{V}_i\}$ for d_q
        end if
    end for
end for
return Visual coreference chains $\{\mathcal{V}_i\}_{i=1}^{N_v}$

A Character Clustering Algorithm

In this section we formally present our incremental clustering algorithm. We also explain how we compute pairwise character matching with a QA method based on LVLMs which we experimentally found to be superior to the more widely used cosine similarity of visual features.

Specifically, we provide crops of two characters to a LVLM and ask whether they refer to the same individual. Since character crops often include background noise (e.g., other characters within the bounding box, such as two characters hugging), we employ fine-grained visual prompting (Yang et al., 2024) to focus the model's attention on the primary character in the crop. We compute visual similarity using LLaVA1.5-13B (Liu et al., 2024a). As this model does not support multi-image input, we concatenate the images of the two characters into a single image. The prompt we used is: *Are the person on the left and the person on the right the same? Output yes or no ONLY.*

B LLM-as-a-Judge Details

We utilized a collection of human evaluation results from four VIST studies: KG-Story (Hsu et al., 2020), PR-VIST (Hsu et al., 2021a), Streth-VIST (Hsu et al., 2021b), and Iter-Blueprint (Liu et al., 2023). Specifically, the first four datasets include annotations for side-by-side overall-level comparisons of two stories, from which we randomly selected 1,000 story pairs. In the Iter-Blueprint dataset, we used the full set of fine-grained annotations for 100 stories, covering aspects such as coherence, engagement, grounding, and overall quality.

Read the following short story and answer the question to identify words referring to people, which includes:

1 . Named entities, including both personal names (e.g., 'John', 'Alice') and specific designations (e.g., 'President', 'CEO').
2. Pronouns as subjects (e.g., he, she, they, I, we) and objects (e.g., him, her, them, us), but excluding possessive forms (e.g., his, our, my, mine).
3. Terms denoting groups of people, including team names, familial terms, and other plural or compound nouns (e.g., 'the Smith family', 'committee', 'USA team').

Context: {{*context*}}

Question:
In the sentence {{*sent*}}, does the word {{*noun*}} refer to people?

Answer (yes or no ONLY):

Table 9: Zero-shot prompt used for detecting character words. ''{{*context*}}'', ''{{*sent*}}'', and ''{{*noun*}}'' are replaced by the story, the sentence that the query words are in, and the query words, respectively.

Image Sequence: {{*Image Sequence*}}

Read the image sequence and write a 5-sentence short coherent story based on the image sequence.

Table 10: Zero-shot prompt used for generating visual stories using GPT-4V. ''{{*Image Sequence*}}'' will be replaced by the input image sequence.

Additionally, the authors randomly selected 100 story pairs from Iter-Blueprint and VIST gold stories for manual annotation, focusing on specificity and character-based side-by-side comparisons. The criteria of each of these metrics can be found in Table 12.

C List of Prompts

This section provides all prompts used in this study. Table 9 presents the zero-shot prompt used for detecting character words. Table 10 provides the zero-shot prompt used for generating visual stories using GPT-4. Table 11 contains the five-shot prompt used for character coreference resolution. Finally, Table 12 displays the zero-shot prompt used for pairwise story comparison.

D VIST++ Dataset: Quality Evaluation

In this section we analyze the quality of our automated annotation pipeline. Recall that we detect character bounding boxes with DETR (Carion et al., 2020) and use OpenCLIP ViT-G/14 (Ilharco et al., 2021) for feature extraction. LLaMA3-70B is used for textual coreference, and SAM (Kirillov et al., 2023) is employed to obtain the segmentation masks for bounding boxes. For LVLM-based pair-wise character matching, we employ LLaVA1.5-13b (Liu et al., 2024a). We evaluate our pipeline against manual annotations provided in VIST-Character (Liu and Keller, 2023) as well as alternative implementations of character detection and coreference modules.

Our results are summarized in Table 1. For textual and visual character detection, we report precision and recall. We assume a textual detection is correct if the head of the noun phrase in question is the same as the gold-standard one. For visual detections, a predicted region is considered correct when the Intersection over Union (IoU; Yu et al. 2016 is higher than a threshold of 50%. For character coreference, we evaluate the coreference chains using B^3 metrics Cai and Strube (2010a), which indicate the average

Annotate all the people mentions within square brackets in the following text with coreference clusters. Use Markdown tags to indicate clusters in the output, with the following format [mention](#cluster_name). For plural nouns, include all related clusters.

Input:
[My friends] and [I] went on a history trip. On the boat, [we] took pictures of other boats like this. [We] used this rope to bring in the anchor for our boat. [One of my partners] posed on the ship for a picture. While the sun went down, [we] cruised and took pictures of the city line.
Output:
[My friends](#1) and [I](#2) went on a history trip. On the boat, [we](#1, #2) took pictures of other boats like this. [We](#1, #2) used this rope to bring in the anchor for our boat. [One of my partners](#1) posed on the ship for a picture. While the sun went down, [we](#1, #2) cruised and took pictures of the city line.

Input:
It is date night for [Tom] and [Susan]. Being sports fans [they] decided to attend a baseball game. [They] are having a great time and being goofy. [Tom] strikes a pose with his sweet ride. Meanwhile, [their kids] are at home with the [babysitter].
Output:
It is date night for [Tom](#1) and [Susan](#2). Being sports fans [they](#1, #2) decided to attend a baseball game. [They](#1, #2) are having a great time and being goofy. [Tom](#1) strikes a pose with his sweet ride. Meanwhile, [their kids](#3) are at home with the [babysitter](#4).

Input:
[James] and [I] were excited to be in Washington D.C. during the 4th of July. There was [a huge crowd of people] already awaiting the firework show. [We] were lucky to find a nice spot on the grass to watch the show. As the evening grew darker [the crowd] was gearing up to enjoy the show, with a great view of the Washington Monument. [I] was able to capture a great photo of the grand finale of the firework show.
Output:
[James](#1) and [I](#2) were excited to be in Washington D.C. during the 4th of July. There was [a huge crowd of people](#3) already awaiting the firework show. [We](#1, #2) were lucky to find a nice spot on the grass to watch the show. As the evening grew darker [the crowd](#4) was gearing up to enjoy the show, with a great view of the Washington Monument. [I](#2) was able to capture a great photo of the grand finale of the firework show.

Input:
[Grandpa] was happy to have [the family] over for the 4th. [His granddaughter] also came for a visit. [They] were so happy to see each other. [They] played for a long time. [Grandpa] took [her] to see some fireworks. [She] was scared of the loud boom, but excited by the bright colors. It was a perfect day.
Output:
[Grandpa](#1) was happy to have [the family](#2) over for the 4th. [His granddaughter](#3) also came for a visit. [They](#1, #2, #3) were so happy to see each other. [They](#1, #2, #3) played for a long time. [Grandpa](#1) took [her](#3) to see some fireworks. [She](#3) was scared of the loud boom, but excited by the bright colors. It was a perfect day.

Input:
Today would be [the baby]'s first trip to the beach. [Daddy] took [the baby] to the beach and showed [him] how the sand and ocean looked. Here the duo are taking a picture snapped by [the mother]. This time [they] all got in on the picture and had a blast! [The baby] had a lot of fun on his first beach trip!
Output:
Today would be [the baby](#1)'s first trip to the beach. [Daddy](#2) took [the baby](#1) to the beach and showed [him](#1) how the sand and ocean looked. Here the duo are taking a picture snapped by [the mother](#3). This time [they](#1, #2, #3) all got in on the picture and had a blast! [The baby](#1) had a lot of fun on his first beach trip!

Input:
{{story}}

Output:

Table 11: The 5-shot prompt used for character coreference resolution. ‘‘{{*noun*}}’’ is replaced by characters previously detected in the input story.

You will be asked to compare two stories generated based on the input images, according to specific criteria. The aim is to assess the quality of each story and select the better one in terms of specificity, coherence, engagement, grounding, characters, and overall preference. Please follow the guidelines below for your evaluation:

Criteria for Evaluation:
1. **Specificity**: Check if the story avoids general statements like "we had a good time" or "I enjoyed myself". A good story should provide detailed, specific examples and descriptions rather than relying on broad, vague statements.
2. **Coherence**: Assess if the story flows smoothly, maintaining logical connections and consistency that make senses to the reader.
3. **Engagement**: Evaluate how interesting the story is, taking into account its unique and memorable features.
4. **Grounding**: Determine how closely the story aligns with the provided input images, capturing their key elements.
5. **Characters**: Check whether the story has clear, recurring characters.
6. **Overall Preference**: Choose the preferred story out of the two provided, based on a combination of specificity, coherence, engagement, grounding, characters and an overall preference.

Evaluation Steps:
1. Carefully read the two stories to grasp the main theme and key points.
2. Compare the two story based on the dimensions provided above (specificity, coherence, engagement, grounding, characters and an overall preference).
3. Construct a dictionary in Python format with your preferred story for each criterion, using 'A' if Story A is better or 'B' if Story B is better. Your feedback should look like this:

```
{
   "specificity": "A" or "B",
   "coherence": "A" or "B",
   "engagement": "A" or "B",
   "grounding": "A" or "B",
   "characters": "A" or "B",
   "overall_preference": "A" or "B"
}
```

Stories:

Story A: {{story_a}}

Story B: {{story_b}}

Output (dictionary ONLY):

Table 12: The zero-shot prompt used for pair-wise story comparison. ''{{*story_a*}}'' and ''{{*story_b*}}'' will be replaced by stories generated by our systems or other models.

percentage of the correctly detected mentions in a chain. For each mention, the B^3 algorithm computes a precision and recall score using the following equations:

$$\begin{aligned} \text{Precision}\,(m_i) &= \frac{|R_{m_i} \cap G_{m_i}|}{|R_{m_i}|} \\ \text{Recall}\,(m_i) &= \frac{|R_{m_i} \cap G_{m_i}|}{|G_{m_i}|} \end{aligned} \tag{2}$$

where R_{m_i} is the chain from the system prediction, which includes the mention m_i, and G_{m_i} is the manually annotated gold-standard chain with m_i. The overall precision and recall are computed by averaging them over all mentions. For multimodal alignment, we use recall and precision.

The first block in Table 13 presents results for textual character detection and coreference resolution, while the second block does the same for visual characters. We observe that our QA-prompting method brings a significant improvement in the precision of textual character mention detection. Previous

Model	Detection			Coreference		
	P	R	F1	P	R	F1
WordNet + SpanBERT	74.0	90.3	81.3	66.6	70.0	68.3
VIST++	**84.9**	**94.5**	**89.4**	**77.2**	**80.3**	**78.7**
CLIP + Kmeans	40.5	69.1	51.1	51.8	60.8	55.9
VIST++	**41.8**	**78.4**	**54.5**	**70.4**	**74.2**	**72.3**

Table 13: Results for textual (upper part) and visual (lower part) character detection and coreference. P and R are short for precision and recall.

Model	Recall	Precision	F1
Liu and Keller (2023)	27.3	27.5	27.4
Our Pipeline	**39.7**	**42.5**	**41.1**

Table 14: Performance of mutlimodal coreference chain alignment. All numbers are percentages (%).

methods (Liu and Keller, 2023; Surikuchi and Laaksonen, 2019) employing WordNet to filter character words often lead to false positives. This is because they operate at the word/phrase level and do not consider contextual semantic information. For example, ‘‘white’’ is usually used to describe color, but its primary sense in WordNet is a person’s name. Also, some character words like ‘‘great-grandmother’’ are not included in WordNet. However, the strong contextual understanding and reasoning capabilities of LLMs can resolve these issues effectively in a question-answering setting.

Our prompting-based method shows approximately a 10% improvement in precision and recall for character coreference. This demonstrates that LLMs are effective coreference resolvers and that structured prompting can predict all sub-clusters referred to by plural nouns, a task that previous methods based on SpanBERT struggled with. It is worth noting that in our experiments, only large LLMs were capable of handling coreference tasks (we used LLaMA3-chat-70B). Smaller models (e.g., LLaMA3-chat-8B) often failed to understand the complex structure of the prompts, resulting in outputs that did not follow instructions.

Our incremental clustering algorithm is superior to previous clustering approaches based on facial features. Using body features reduces the number of missing characters, especially when their faces are not visible in the images. Additionally, our algorithm addresses a critical limitation of previous clustering approaches: the same person cannot appear more than once in a single image. These improvements effectively enhance the accuracy of visual character coreference, resulting in a 10%–15% increase in the accuracy of multimodal chain alignment, as shown in Table 14. Overall, our results demonstrate that our pipeline achieves state-of-the-art performance in character coreference resolution.