Technical Summary: PG-STORY: Taxonomy, Dataset, and Evaluation for Ensuring Child-Safe Content for Story Generation

Kai-Yu Lu

1 Research Problem and Motivation

1.1 Research Problem

The paper studies how to ensure that automatically generated narrative text, in particular short stories, is safe and appropriate for children under the age of ten. The study focuses on content safety in neural story generation systems and introduces both an annotation taxonomy and a model-based framework that can diagnose and reduce unsafe content in generated stories.

1.2 Motivation and Practical Context

Large language models such as ChatGPT, LLaMA, and PaLM 2 have made it straightforward to generate long, coherent narratives on demand. For children, automatically generated stories can in principle provide personalized and engaging reading material. However, these models are typically trained on general web-scale text without explicit child-centric safety constraints. As a result, generated stories may contain problematic elements such as graphic violence, sexual content, offensive language, or discriminatory descriptions that are unsuitable for young readers.

In contrast to movies, television, and video games, which are governed by standardized rating systems such as TV parental guidelines and game rating boards, text-based digital content lacks consistent and machine-usable content ratings. Existing natural language processing research has concentrated on toxic language and abusive content in social media comments and online conversations, with datasets focusing on insults, threats, and hate speech. These resources are not directly aligned with the needs of child-safe story generation for three reasons.

First, social media comments are short and conversational, whereas stories are multi-sentence narratives with plots, characters, and atmosphere. Harmful content in stories can appear through cumulative context and narrative implications rather than isolated offensive words. Second, existing toxicity datasets often cover only a subset of safety dimensions, such as profanity or hate speech, and do not explicitly consider child-centric concerns such as frightening scenarios or substance abuse in narrative form. Third, publicly available evaluation tools such as Perspective API and Detoxify are trained on adult-centric notions of offensiveness and may misclassify content that is harmless for adults but disturbing for children, or vice versa.

The paper is motivated by this mismatch and by the increasing time that children spend consuming digital content. The central motivation is to provide a principled way to define, detect, and mitigate unsafe textual content in stories that are intended for young readers.

1.3 Research Gaps

The study identifies several concrete gaps in prior work.

- There is no standardized, child-centric taxonomy for content safety in narrative text that covers multiple types of problematic content relevant to children under ten years old.
- There is no benchmark dataset that combines story-level and sentence-level safety annotations for narratives, which is necessary to train and evaluate child safety classifiers.
- Existing toxic language detectors are not calibrated to child safety in narrative contexts, and their limitations are not systematically quantified against a child-focused taxonomy.
- There is no story generation framework that jointly performs story planning, safety self-diagnosis, and controlled rewriting of unsafe sentences to obtain child-safe outputs.

The PG-STORY project is designed to close these gaps by providing a taxonomy, datasets, classifiers, and a safe story generation framework.

2 Related Work

2.1 Toxic Language and Abusive Content Detection

A large body of research studies offensive or abusive language in online platforms. Representative datasets include the Contextual Abuse Dataset for Reddit comments, cyberbullying datasets from Twitter, toxic comment datasets from Wikipedia talk pages, and abusive language datasets from YouTube comments. These corpora typically provide sentence-level labels such as toxic, abusive, hate, or insult, sometimes accompanied by target information such as the group being attacked.

These resources have enabled the development of classifiers based on pre-trained encoders such as BERT. These models can detect explicit offensive language, but they are trained on short messages and adult-centric notions of offensiveness, and they do not explicitly encode a child safety perspective or narrative context.

2.2 Child Safety in Multimedia Content

Prior work on child safety has largely focused on multimedia platforms such as YouTube. Studies have examined disturbing cartoons, inappropriate videos targeting children, and abusive comments associated with child-oriented videos. Models have been trained to detect inappropriate visual scenes, violent audio, or harmful comments accompanying videos. These approaches are valuable but focus on video and audio modalities rather than long-form narrative text.

2.3 Story Generation and Controlled Text Generation

Neural story generation has been studied through hierarchical decoders, content planning strategies, and plan-and-write frameworks. The plan-and-write paradigm first constructs a high-level plan, such as a sequence of keywords or plot points, and then expands this plan into a full story.

Controlled text generation research has introduced methods for steering language models toward desired attributes such as sentiment or style. Representative approaches include conditional Transformer models where control codes are prepended to the input, Plug-and-Play Language Models (PPLM) which adjust hidden states using an auxiliary attribute model, and weighted decoding schemes that modify next-token probabilities according to attribute constraints.

These methods demonstrate that generation can be guided by attribute models, but they have rarely been applied specifically to child safety, and they have not been combined with explicit self-diagnosis of safety at sentence and story level.

2.4 Existing Story and Children's Text Corpora

Several corpora contain stories that are suitable for children, such as Children Stories Text Corpus and Children's Book Test. These collections are drawn from sources such as Project Gutenberg and focus on language modeling or reading comprehension tasks rather than content safety annotations.

ROCStories is a widely used dataset of five-sentence commonsense stories that cover everyday events. FAIRYTALEQA and similar datasets present question answering tasks over classic fairy tales. None of these corpora include explicit annotations of unsafe categories for children. Therefore, they cannot be used directly to train or evaluate child-centric safety models.

3 Dataset Construction

3.1 Child Safety Taxonomy for Narrative Text

The paper introduces a child-centric safety taxonomy for narrative text intended for children under ten years old. The taxonomy defines five disjoint categories of potentially harmful content. Each category is intended to capture a distinct safety dimension with minimal overlap.

- **Profanity & Slurs**. Contains swear words, taboo expressions, and insulting slurs. This includes explicit obscene terms and rude language directed toward individuals or groups.
- **Sex & Nudity**. Contains sexual activities, sexual references, implied sexual behavior, or descriptions of nudity that are inappropriate for young children.
- **Violence & Scariness**. Contains violent actions, self-harm, references to weapons, risky or unhealthy dangerous activities, or intense frightening scenes such as horror elements.
- **Substance Consumption**. Contains references to alcohol, tobacco, drugs, or other substances, including explicit or implied consumption and abuse.
- **Discrimination & Bias**. Contains explicit or implicit insults, derogatory statements, or discriminatory attitudes concerning age, ethnicity, gender, religion, appearance, ideology, or socio-economic status.

The taxonomy is informed by guidelines from child advocacy organizations and by national content rating standards used in other media, but it is tailored specifically to narrative text.

3.2 Unsafe Content Corpus (UNSAFECORPUS)

To train specialized safety classifiers that operate under the new taxonomy, the authors construct an unsafe content corpus named UNSAFECORPUS by combining several existing datasets spanning four major online platforms. The corpus includes both safe and unsafe texts, with unsafe instances determined by offensive labels in the source datasets.

Not all categories of the child safety taxonomy are explicitly labeled in the original datasets. To enrich category coverage, the authors compile harmful lexicons from several sources, including profane word lists and a children's taboo lexicon. After removing words that frequently occur in non-offensive contexts, approximately 1,690 lexicon entries are manually aligned with the five safety categories.

Lexicon matching is then used to identify additional unsafe instances in each category. For instance, the number of samples containing profanity and slurs increases from around one thousand without lexicon matching to roughly thirty-nine thousand with lexicon matching. Violence and scariness as well as substance consumption also receive substantial increases in coverage.

Source Dataset	Platform	Safe Samples	Unsafe Samples
Contextual Abuse Dataset	Reddit	13,577	9,618
Cyberbullying Dataset	Twitter	0	46,017
Toxic Comment	Wikipedia talk pages	84,000	42,778
Unsafe Transcription	YouTube	258	98
Total		97,815	98,511

Table 1: Data distribution for UNSAFECORPUS, aggregated over four platforms.

Dataset	Typical Length	Writer Type	# Stories	# Sentences
ROCStories	Short	Crowd-sourced	52,665	263,325
WikiPlots	Long	Crowd-sourced	112,936	$\approx$ 1,000,000
FAIRYTALEQA narratives	Long	Expert	278	26,208
Grimm's Fairy Tales	Long	Expert	115	5,348

Table 2: Story sources underlying the PG-STORY corpus before sampling and segmentation.

UNSAFECORPUS is split into training, validation, and test sets with a ratio of 60 percent, 20 percent, and 20 percent respectively, ensuring that both safe and unsafe instances are represented in each split.

3.3 PG-STORY Narrative Corpus

The main contribution on the data side is PG-STORY, a corpus of stories annotated with child safety labels at both sentence and discourse levels. The corpus merges narratives from four sources that vary in length, authorship, and style.

Long narratives from WikiPlots, FAIRYTALEQA, and Grimm's Fairy Tales are segmented into shorter passages that are typically around five sentences, so that each passage can be treated as a single story instance for safety annotation. ROCStories stories are already short, so they are used as-is.

PG-STORY includes one thousand short stories or story excerpts that are fully annotated by human annotators for both sentence-level and discourse-level safety. An additional one hundred thousand instances receive semi-supervised labels from the specialized safety classifier.

3.4 Model-in-the-loop Annotation Protocol

To improve annotation efficiency and focus human effort on informative examples, the authors adopt a model-in-the-loop strategy. A pre-trained BART-based detection model, trained on UNSAFECORPUS, is first applied to all candidate story sentences and passages.

For each sentence, the detection model outputs a safety score between zero and one, where scores closer to one indicate higher predicted risk. Sentence scores are aggregated into a discourse-level safety score for each passage, for example by averaging. These scores are then used to stratify the pool of candidate stories into different risk bands.

In the first iteration, 125 stories from each source dataset are sampled across risk bands, resulting in 500 stories that are manually annotated on Amazon Mechanical Turk by native English speakers from selected countries. These annotations are used to refine the detection model, and a second iteration repeats the process for another 500 stories. The final PG-STORY corpus thus contains 1,000 human-annotated stories plus a larger set of semi-supervised annotations.

3.5 Human Annotation Protocol and Agreement

Annotators judge both discourse-level and sentence-level safety under the assumption that the audience consists of children under the age of ten. For each story, annotators answer whether the story is overall safe, unsafe, or uncertain. If unsafe, annotators indicate which of the five categories apply. The same decisions are then collected for each sentence.

Inter-annotator agreement is measured using Cohen's kappa and Fleiss' kappa. Both sentence-level and discourse-level annotations achieve average kappa scores around 0.26 to 0.27, reflecting moderate agreement given the subjectivity of safety judgments. Category frequencies show that violence and scariness occur most often among unsafe stories, followed by discrimination and bias, as is typical for fairy tales and plot summaries.

4 Query Protocol and Task Definitions

4.1 Unsafe Content Detection Task

The first task is binary detection of unsafe content according to the child safety taxonomy. The input is a sentence or short text segment from UNSAFECORPUS. The output is a label *safe* or *unsafe*. Detection models are evaluated using precision, recall, and F1 score computed separately for safe and unsafe classes, as well as macro averages.

Perspective API and Detoxify provide a continuous toxicity score between zero and one. For evaluation, a threshold of 0.5 is used: scores greater than or equal to 0.5 are mapped to the unsafe class and lower scores to the safe class.

4.2 Unsafe Category Classification Task

The second task is multi-class categorization of unsafe content into one or more categories from the five-category taxonomy. For sentences labeled unsafe, a classifier predicts which categories apply. The specialized BART-based categorization model is trained with a multi-class output layer that maps hidden representations to category logits.

Since existing tools such as Perspective API and Detoxify do not directly output the five categories, their performance is approximated by measuring recall within each category. A sample is treated as detected if its toxicity score exceeds the threshold, and recall is computed per category.

4.3 Child Safety Assessment for Language Model Outputs

The study also evaluates the ability of a general-purpose conversational model, ChatGPT 3.5, to identify unsafe sentences. For this test, one hundred sentences are sampled from the UNSAFECORPUS test set, with twenty sentences from each unsafe category. ChatGPT is prompted with an instruction of the form: read a sentence and output 0 if safe and 1 if unsafe.

Accuracy, precision, recall, and F1 score of ChatGPT predictions are computed against the ground truth unsafe labels. Category-wise recall is computed in the same way as for other tools.

In addition, the specialized safety model is applied to 85 stories generated by ChatGPT in response to prompts that request a short story for kids but use adjectives that attempt to bypass safety restrictions. The fraction of stories flagged as unsafe provides an empirical estimate of residual risk.

4.4 Story Generation and Safety Evaluation Tasks

For story generation, the input consists of a story title and a set of content keywords. Keywords are automatically extracted from human-written stories in PG-STORY using Rapid Automatic Keyword Extraction (RAKE), an unsupervised keyword extraction algorithm that identifies salient phrases based on co-occurrence and word frequency.

The words are combined into a flattened sequence containing the title, a special end-of-title token [EOT], the comma-separated keyword list, and an end-of-plan token [EOP]. The generation model outputs a sequence of sentences that form the story. In the conditional generation setting, each sentence is prefixed by special tokens that indicate safety and category information. In the self-diagnosis setting, each sentence is followed by corresponding safety tokens that the model predicts.

Generated stories are evaluated along two dimensions.

- **Linguistic quality and semantic faithfulness**. This includes perplexity, BERT-based semantic similarity with reference stories, diversity metrics, and the keywords matching ratio.
- **Content safety**. This includes automatic toxicity scores from Perspective API and human safety judgments for discourse-level stories and sentence-level components.

5 Modeling Approach

5.1 Safety Detection and Categorization Models

Both detection and categorization models are built on top of BART, a sequence-to-sequence Transformer architecture that combines a bidirectional encoder with an auto-regressive decoder. The encoder processes input sequences into contextual vector representations, and the decoder predicts output tokens conditioned on encoder states and prior outputs.

For the binary detection task, the model uses a pre-trained BART encoder and attaches a non-linear projection with dropout followed by a binary classification layer. The final hidden state corresponding to a special classification token is passed through this classification head to produce a scalar logit, which is mapped to a probability via a logistic function. This probability represents the estimated risk that the input text is unsafe.

For the multi-class categorization task, the architecture is similar, except that the classification head outputs a vector of logits corresponding to the five unsafe categories. A softmax function converts these logits into category probabilities. During training, cross-entropy loss encourages the model to assign higher probability to correct categories.

5.2 Plan-to-Story Generation Model

The story generation component uses BART as a conditional generative model. The encoder receives the flattened combination of title, [EOT], keyword list, and [EOP]. The decoder initializes its state and generates story sentences token by token.

During training, teacher forcing is used: the decoder conditions on the ground truth story tokens. The model learns to map high-level plans in the form of keywords to coherent narratives. This plan-to-story setup follows the general plan-and-write paradigm, but is instantiated with the chosen special tokens for title and plan boundaries.

5.3 Conditional Generation with Safety Tokens

In the conditional generation setting, each target sentence in the training data is prefixed with safety control tokens. The first token indicates overall safety status: [SAFE] for sentences judged safe and [UNSAFE] for sentences judged unsafe. For unsafe sentences, additional category tokens [1] to [5] indicate which unsafe categories are present, aligned with the taxonomy.

The flattened input to the decoder therefore starts with safety tokens, followed by the sentence tokens and an end-of-sentence token [EOS]. The model learns to associate certain lexical and syntactic patterns with the control tokens. At inference time, the model can be conditioned on [SAFE] to encourage the generation of safe sentences, or on particular category tokens to synthesize examples containing specific types of unsafe content for analysis.

5.4 Safety Self-Diagnosis

In the self-diagnosis setting, the model architecture is unchanged, but the sequence layout during training is different. Each generated sentence is followed by its own safety annotation, so that the decoder outputs a sentence, an [EOS] token, and then safety tokens such as [SAFE] or [UNSAFE] [3]. At the end of the entire story, the decoder emits an [END_STORY] token followed by discourse-level safety tokens.

This design forces the model to produce safety judgments as part of the same decoding process used for storytelling. Intuitively, the model first generates content and then classifies its own content. During inference, the model can write a story one sentence at a time and immediately attach a predicted safety label to each sentence and to the full story.

5.5 Content Re-write via Controlled Generation

The content re-write module is triggered whenever the self-diagnosis output for a sentence contains [UNSAFE]. The goal is to replace the unsafe sentence with an alternative that preserves story coherence while eliminating or mitigating unsafe elements. Two controlled generation techniques are investigated.

Plug-and-Play Language Model (PPLM). PPLM is a method that steers the hidden states of a pre-trained language model toward or away from certain attributes. An auxiliary attribute model, in this case the specialized safety classifier, assigns a probability that a partially generated sentence is unsafe. During decoding, gradients from the attribute model are used to modify the hidden state before sampling the next token, reducing unsafe probability and increasing safe probability.

In this framework, the original unsafe sentence is taken as context, and PPLM generates an alternative continuation that has lower predicted risk according to the safety classifier. Over multiple steps, the process yields a rewritten sentence that is more likely to be safe for children.

Weighted Decoding. Weighted decoding modifies next-token probabilities by combining the base language model probabilities with attribute scores. For each candidate token, the attribute model evaluates the result of appending that token to the current sequence. The combined score is a weighted sum of the language model log probability and an attribute-based adjustment. Tokens that reduce predicted unsafety receive higher combined scores, while tokens that increase unsafety are penalized.

Weighted decoding is computationally simpler than PPLM because it does not rely on gradient updates to hidden states, but it may have less flexibility in exploring diverse safe rewrites.

5.6 Hyperparameters and Implementation Details

The models rely on public BART implementations available in modern Transformer libraries. The paper states that the plan-to-story model and the conditional or self-diagnosis models are fine-tuned on the combined story datasets from Table 2, with PG-STORY providing safety annotations. The main text does not report detailed numerical values for hyperparameters such as learning rate, batch size, or number of training epochs. These details are described as following standard fine-tuning practices and are provided in the released code.

To align with the technical detail guideline, the main hyperparameter configuration can be summarized as follows.

- Base architecture: BART encoder-decoder with standard hidden size and number of layers.
- Optimization: Transformer-style fine-tuning with cross-entropy loss on token predictions and classification heads.
- Safety classifier training: supervised learning on UNSAFECORPUS for detection and categorization tasks.
- Story generation training: supervised learning on plan-story pairs with or without safety tokens, using PG-STORY splits.

5.7 Experimental Environment

The main text does not specify hardware details such as GPU type, memory configuration, or training time. The experiments are implemented using a common deep learning framework that supports Transformer-based models. Since the emphasis is on taxonomy, data, and modeling strategies rather than computational performance, the environment is not a central focus of the paper.

6 Empirical Results

6.1 Detection Performance on UNSAFECORPUS

The detection experiment compares the specialized child safety detection model with Perspective API and Detoxify on the UNSAFECORPUS test set. A sample is classified as unsafe if its toxicity score is at least 0.5 for the external tools, while the specialized model directly outputs safe or unsafe labels.

For safe content, Perspective API and Detoxify achieve similar precision around 62 percent and extremely high recall around 99 percent, which means that they rarely miss safe content but often misclassify safe content as unsafe. For unsafe content, both tools achieve precision around 98 percent but recall around 41 percent, indicating that they correctly flag unsafe content when they do so, but fail to detect a large fraction of unsafe instances.

The specialized detection model obtains precision around 95.6 percent and recall around 96.1 percent on safe content, and precision around 98.1 percent and recall around 97.8 percent on unsafe content. Overall macro F1 score is approximately 96.9 percent, substantially higher than the roughly 67 percent macro F1 scores of Perspective API and Detoxify. This indicates that a domain-specific child safety model can simultaneously avoid over-blocking safe content and under-detecting unsafe content.

6.2 Category-level Classification Performance

Category-level results show that the specialized categorization model achieves F1 scores above 89 percent for four of the five categories. Approximate F1 scores are 93.3 percent for profanity and slurs, 89.1 percent for sex and nudity, 86.2 percent for violence and scariness, and 93.6 percent for discrimination and bias. Substance consumption has a lower F1 score around 49.9 percent due to the small number of training examples in this category.

When recall is compared across models, the specialized categorization model significantly outperforms Perspective API and Detoxify in all categories. For example, in violence and scariness, the specialized model reaches recall around 82.4 percent, while Perspective API and Detoxify remain near 24 percent. Substance consumption recall is around 50 percent for the specialized model, higher than the around 31 to 32 percent achieved by external tools. This analysis confirms that general-purpose toxicity tools leave many child-relevant unsafe categories insufficiently covered, particularly violence, scariness, and substance consumption.

6.3 Child Safety Assessment for ChatGPT

On the one hundred test sentences sampled from UNSAFECORPUS, ChatGPT 3.5 achieves accuracy around 72.6 percent, precision around 72.4 percent, recall around 72.8 percent, and F1 score around 72.6 percent for unsafe detection. These numbers exceed the performance of Perspective API and Detoxify on the same test subset, which achieve accuracies around 64.5 to 64.6 percent and F1 scores around 63.4 to 63.5 percent.

Category-wise recall shows that ChatGPT is more capable than general tools in detecting profanity and discrimination, but still lags the specialized model. For example, recall for profanity and slurs is around 0.80 for ChatGPT and 0.75 for the specialized model, while recall for sex and nudity is around 0.73 for ChatGPT and 0.85 for the specialized model. For violence and scariness, ChatGPT reaches recall around 0.63, significantly higher than the 0.24 recall of external tools, but lower than the 0.87 recall of the specialized model. Substance consumption recall remains low for ChatGPT at around 0.32, whereas the specialized model reaches approximately 0.98 on this subset.

In a separate experiment, 85 children-oriented prompts with deliberately adversarial adjectives are submitted to ChatGPT to generate short stories. The specialized safety model flags approximately 52 percent of these stories as unsafe for children, demonstrating that even with prompting that mentions children, existing large language models can produce content that violates the taxonomy.

6.4 PG-STORY Annotation Statistics

Human annotations on PG-STORY illustrate the distribution of unsafe content across data sources. For discourse-level labels, a sizeable portion of stories from WikiPlots and Grimm's fairy tales is judged unsafe, mainly due to violence and scariness. For instance, WikiPlots and Grimm's fairy tales contain hundreds of stories with violent or frightening content, which reflects typical narrative patterns in classical literature.

Unsafe categories occur in combinations. Violence and scariness accounts for the majority of discourse-level unsafe labels, followed by discrimination and bias and then profanity and sex-related content. Substance consumption is relatively rare but present, especially in modern plots or adult-oriented story summaries.

6.5 Story Generation Quality and Safety

Automatic metrics for story generation and content re-write highlight trade-offs between quality, diversity, and safety.

Perplexity values around 1.589 to 1.591 for plain story generation (self-diagnosis and conditional generation) indicate fluent text according to the language model. Self-diagnosis and conditional generation attain similar BERT-based semantic similarity scores with reference stories, around 0.812 to 0.816, reflecting comparable semantic faithfulness.

Diversity metrics Dist-1, Dist-2, and Dist-3, which measure the proportion of distinct unigrams, bigrams, and trigrams, are higher for self-diagnosis than for conditional generation. Dist-1 increases from approximately 0.134 to 0.166, with similar improvements for higher order n-grams. This suggests that self-diagnosis introduces slightly more lexical variety, possibly because the model is not constrained by fixed control tokens at sentence beginnings.

For content re-write, both PPLM and weighted decoding substantially increase diversity. Dist-1 and Dist-2 reach values above 0.47 and 0.89 respectively, indicating that rewrites introduce many new wording patterns. At the same time, perplexity increases to around 7.4 to 8.5, reflecting that rewritten stories deviate more strongly from reference texts and are therefore less predictable under the base language model.

The keywords matching ratio, which measures how many extracted keywords appear in the generated story, decreases from approximately 0.71 for plain generation to around 0.47 to 0.49 for rewritten stories. This is expected, because unsafe keywords that trigger rewriting often correspond to problematic concepts and are explicitly removed or replaced.

Toxicity scores computed by Perspective API show that both PPLM and weighted decoding reduce predicted unsafety compared to plain generation. Toxicity drops from around 0.168 to approximately 0.123 for PPLM and to around 0.143 for weighted decoding. This indicates successful mitigation of unsafe content according to an external evaluation tool.

6.6 Human Evaluation of Safety and Rewriting

Human evaluation is conducted on thirty unseen test stories. Each story is paired with outputs from four configurations: self-diagnosis, conditional generation, self-diagnosis plus PPLM re-write, and self-diagnosis plus weighted decoding re-write. Annotators judge whether each generated story and each sentence is safe for children.

For safety prediction accuracy, self-diagnosis attains higher discourse-level agreement with human judgments than conditional generation. Discourse-level safety prediction accuracy is about 63.3 percent for self-diagnosis, compared to 40.0 percent for conditional generation. At the sentence level, both methods show similar accuracy, around 73.4 percent for self-diagnosis and 72.5 percent for conditional generation.

Re-writing success is defined as the proportion of unsafe sentences that become safe after rewriting according to human evaluators. PPLM achieves a re-write success rate of about 54.5 percent, whereas weighted decoding achieves about 27.2 percent. At the sentence level, PPLM converts approximately 48.7 percent of unsafe sentences into safe ones, while weighted decoding succeeds for about 25.6 percent. These results indicate that PPLM, despite being more complex, is more effective at generating child-safe alternatives.

7 Summary

7.1 Key Contributions and Novelty

The PG-STORY study contributes a complete pipeline for child-safe story generation in three main aspects.

First, the study proposes a child-centric safety taxonomy for narrative text that includes five categories covering profanity and slurs, sex and nudity, violence and scariness, substance consumption, and discrimination and bias. This taxonomy is tailored to children under ten years old and designed for application to multi-sentence narratives.

Second, the study constructs two complementary datasets. UNSAFECORPUS integrates several offensive language datasets and enriches them with lexicon-based category labels to train strong safety detectors. PG-STORY merges multiple story sources into a corpus of one thousand human-annotated stories and one hundred thousand semi-supervised stories with sentence-level and discourse-level safety labels, creating a benchmark for evaluating content safety in story generation.

Third, the study introduces a safe story generation framework that combines a plan-to-story model, a safety self-diagnosis component, and a content re-write module based on controlled generation techniques such as PPLM and weighted decoding. This framework allows a model to generate stories, assess its own outputs, and rewrite unsafe sentences using an attribute model grounded in the safety taxonomy.

Compared with prior work, the novelty lies in reframing content safety as a child-centric, narrative-focused problem, providing a structured taxonomy and datasets, and demonstrating self-diagnosis and rewriting within a single story generation architecture.

7.2 Limitations

The study acknowledges several limitations.

- Age granularity is coarse. All judgments are made with respect to children under ten years old as a single group. Developmental differences between toddlers, younger children, and pre-teens are not explicitly modeled.
- Cultural variation is not fully considered. Annotators are drawn from English-speaking countries, and cultural norms about what is safe for children differ across societies. The dataset may reflect specific cultural assumptions.
- Semi-supervised labels depend on model predictions. Many PG-STORY instances are annotated by the safety classifier rather than by humans, which may propagate model biases.
- Substance consumption remains challenging. Sparse data in this category leads to lower classification performance, suggesting the need for more targeted data collection.
- Computational and deployment aspects are not explored in depth. The framework requires multiple passes of detection and rewriting, which may be expensive for very long narratives.

7.3 Future Directions

Future work can extend the PG-STORY framework in several directions.

- Age-sensitive safety modeling that distinguishes several child age groups and assigns differentiated safety thresholds.
- Cross-cultural annotation campaigns that include annotators from diverse cultural backgrounds to better capture global norms.
- Improved modeling of low-resource categories such as substance consumption through targeted data collection and specialized lexicons.
- Integration of multimodal signals, such as illustrations or audio, for picture books or narrated stories.
- Closer coupling with general-purpose large language models to enable real-time safety monitoring and rewriting in interactive storytelling systems.

7.4 High-level Takeaways

From a technical perspective, the main lessons of the PG-STORY study can be summarized as follows.

- Child-centric safety for story generation requires an explicit taxonomy and dedicated datasets. General toxic language tools are not sufficient to guarantee child safety.
- Specialized safety classifiers trained under the proposed taxonomy substantially outperform general-purpose toxicity detectors and generic language models in both binary detection and category-level recall.
- A generation framework that combines planning, self-diagnosis, and controlled rewriting can meaningfully reduce unsafe content in generated stories while maintaining reasonable fluency and diversity.