MLD-EA: Check and Complete Narrative Coherence by Introducing Emotions and Actions

Jinming Zhang Yunfei Long
University of Essex
jz22273@essex.ac.uk yl20051@essex.ac.uk

Abstract

Narrative understanding and story generation are critical challenges in natural language processing (NLP), with much of the existing research focused on summarization and question-answering tasks. While previous studies have explored predicting plot endings and generating extended narratives, they often neglect the logical coherence within stories, leaving a significant gap in the field. To address this issue, we introduce the **M**issing **L**ogic **D**etector by **E**motion and **A**ction (**MLD-EA**) model, which leverages large language models (LLMs) to identify narrative gaps and generate coherent sentences that integrate seamlessly with the story's emotional and logical flow. The experimental results demonstrate that the MLD-EA model enhances narrative understanding and story generation, highlighting LLMs' potential as effective logic checkers in story writing with logical coherence and emotional consistency. This work fills a gap in NLP research and advances border goals of creating more sophisticated and reliable story-generation systems.

1 Introduction

Narrative understanding and story generation have been a compelling challenge in Natural Language Processing (NLP) for a long. They evolved from early rule-based systems with limited creativity to sophisticated models that generate rich, engaging narratives (Mooney and DeJong, 1985; Fan et al., 2018). Introducing Transformer (Vaswani, 2017) models like BART (Lewis et al., 2020) and large language models (LLMs) like ChatGPT (OpenAI, 2022) revolutionized this task by utilizing advanced architectures to capture in-detailed dependencies.

Many previous studies have focused on tasks like summarizing (Awasthi et al., 2021; Jin et al., 2024), sentiment analysis (Lu et al., 2023; Zhao et al., 2025; Lu et al., 2025) and question-answering (QA) (Zhuang et al., 2024; Huang et al., 2024a). While previous story generation research often centered on predicting plot endings or crafting long narratives (Guan et al., 2020; Li et al., 2022). However, in general, story writing frequently needs to pay more attention to maintaining logical coherence (Oatley, 2002; Currie and Jureidini, 2004).

Not surprisingly, some recent works lead LLMs to maintain narrative coherence in different ways with effective results (Zhao et al., 2023; Wang et al., 2023). However, most of those works focus on continuously writing coherency stories by LLMs (Guan et al., 2021). There is still a gap in detecting the logical coherence in the narratives.

To address this gap, our approach focuses on the observable actions of characters rather than delving into their deeper motivations. This choice stems from the understanding that actual actions have a more immediate and direct impact on emotions, and conversely, emotions are often the driving force behind tangible actions (Zhu and Thagard, 2002; Döring, 2003). The James-Lange theory of emotion in psychology posits that physiological responses to a situation—such as a racing heart or clenched fists—occur first and then lead to the subjective experience of emotion (Cannon, 1927). This suggests that an observable action (like a person slamming a door) can directly trigger an emotional response (such as anger or frustration). Similarly, the cognitive-behavioral theory emphasizes that behaviors (actions) and emotions are closely linked, where a behavior change can directly influence emotional states, and vice versa (Maslow, 1943; Eisenberg, 2014; Leahy et al., 2022).

By prioritizing the direct interplay between observable actions and emotions, we aim to capture the essence of narrative logic in a way that reflects these well-documented psychological principles (Carver et al., 2000). This approach is supported by extensive psychological studies that emphasize the strong correlation between actions and emotional responses, such as how consistent patterns of behavior can shape long-term emotional states,

Proceedings of the 31st International Conference on Computational Linguistics, pages 1892–1907
January 19–24, 2025.

as seen in theories of learned helplessness or social learning (Bandura, 1977).

In this study, we introduce the **M**issing **L**ogic **D**etector by **E**motion and **A**ction (MLD-EA), a LLM-based model designed to identify gaps in narrative logic and generate missing plot elements that are coherent both logically and emotionally. By incorporating the relationship between actions and emotions, MLD-EA aims to enhance the logical structure of narratives. Experimental results demonstrate that our models can produce more believable and emotionally coherent stories by aligning narrative generation with these psychological insights. Our model improves narrative understanding and story generation, underscoring the potential of LLMs as story generators and powerful logic checkers in the creative process.

The main contributions of our work can be briefly summarized as follows: **1)**, We propose a novel task of narrative logic detection. **2)**, By grounding our model in cognitive-behavioral theories, we highlight how emotions directly interact with actions, leading to better narrative understanding and generation. **3)**, Experiments have shown that our MLD-EA model has achieved superior results in most aspects, including narrative logic checking with involved characters' emotions and actions and missing plot completeness. Also, we demonstrate the importance of behavior and emotion in story logic detection and generation.

Leveraging this interaction between actions and emotions to assess and generate story logic more efficiently and accurately mirrors the natural cause-and-effect relationships in human behavior.

2 Related Works

Several innovative approaches have been developed to enhance AI-generated narratives' logical coherence, emotional depth in narrative understanding, and story generation within NLP. Paul and Frank (2021) framework introduces a recursive inference strategy that dynamically generates contextualized rules to guide narrative completion, focusing on maintaining coherence and logical flow throughout the story. Similarly, the CHAE model (Wang et al., 2022) offers fine-grained control over narrative elements, creating customized stories with specific characters, actions, and emotions, enhancing the personalization and richness of the narratives. Similarly, the COMMA (Xie et al., 2022) explores the relationships among motivations, emotions, and

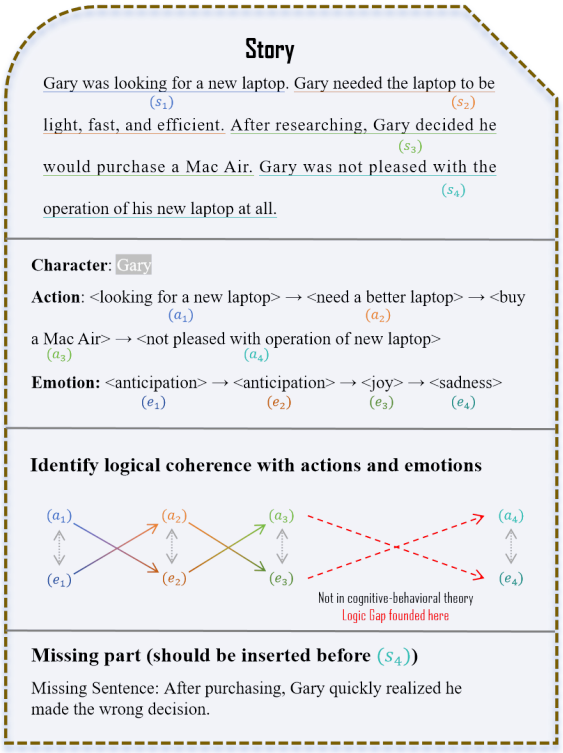

Figure 1: A task example. "Identify logical coherence with actions and emotions" is checking the logical coherence guided by the cognitive-behavioral theory.

actions, providing a cognitive framework that deepens the understanding of narrative construction by modeling these interrelated factors. However, these traditional models often struggle to consistently integrate actions and emotions to maintain logical coherence throughout the entire narrative, leading to disjointed or emotionally inconsistent storylines when handling more complex plots (Kambhampati et al., 2024). Additionally, they may lack the flexibility to dynamically understand nuanced shifts in a character's behavior or emotional progression.

Exploring LLMs, cognitive frameworks, and hybrid planning strategies has paved the way for more engaging and human-like stories. Alvarez (2023) used ChatGPT in interpreting narrative structures, which further extends the potential for generating stories based on predefined structures, offering new methods for narrative development. Notably, approaches such as iterative prompting-based planning for suspenseful story generation (Xie and Riedl, 2024), the combination of symbolic planning with neural models (Farrell and Ware, 2024), and the SWAG method (Patel et al., 2024), which utilizes action guidance in storytelling, have significantly improved the quality and engagement of AI-generated narratives. Additionally, compre-

hensive evaluations like "The Next Chapter" (Xie et al., 2023) and knowledge-enhanced pre-training models (Guan et al., 2020) have shown that LLMs can produce stories of high quality, sometimes approaching the level of human authors. LLMs often struggle to maintain consistent plots on generation, but they cannot check their generated stories by themselves (Huang et al., 2024b). In our approach, MLD-EA is able to find such logical loopholes by introducing the interaction between emotions and actions to keep stories coherent.

3 Problem Definition

The primary goal of MLD-EA is to identify whether the input story is logically completed, as Figure 1 shows. We divided the model into four main sub-tasks: **1)**, abstracting characters' actions. **2)**, classifying their emotions for each sentence. **3)**, then locate the logical loopholes of the narrative in which the missing part should be inserted. **4)**, we complete the tale consistently by predicting the characters' actions and emotions. Thereby preserving the narrative's overall coherence and logical structure. The tasks are defined as follows:

For any input n sentences story ($S = s_1, \cdots, s_n$) with m characters appeared in this story ($C = c_1, \cdots, c_m$), MLD-EA abstract characters' actions a and classify their emotions e for each sentence, denoted as $\{(c, s) \rightarrow (a(c, s), e(c, s)) \mid c \in C, s \in S\}$, where $a(c, s)$ represents the action of character c in sentence s and $e(c, s)$ represents the emotion of character c in sentence s.

Sequently, given the story and characters' actions and emotions, MLD-EA will use the provided information to review the story and find inconsistencies. Notably, our task is to find the logic gap in the inner story. We suppose the start and end of the story are always complete. The process involves identifying points where the characters' actions or emotions exhibit abrupt changes that the preceding context cannot logically explain. After that, MLD-EA outputs the index k which the missing part should be inserted before it:

$$k = \begin{cases} 1 < k < n & \textit{if there is a} \\ & \textit{missing sentence} \\ -1 & \textit{otherwise}. \end{cases} \quad (1)$$

Formally, if MLD-EA identifies a logic gap before a specific place k in the story, it proceeds by predicting the most likely actions $\hat{a}(c, s_k)$ and emotions $\hat{e}(c, s_k)$ by using the sequence of preceding ($\{a(c, s_{k-1}), e(c, s_{k-1})\}$) and succeeding actions and emotions ($\{a(c, s_k), e(c, s_k)\}$). Then MLD-EA estimates the most coherent missing sentence s_k according to $\hat{a}(c, s_k)$ and $\hat{e}(c, s_k)$.

4 Methodology

In this section, we will provide a detailed methodology for each module within our MLD-EA model. The model architecture is shown in Figure 2.

4.1 Action Abstraction

The action abstraction module is designed to extract and abstract actions performed by characters in a given sentence, playing a crucial role in analyzing narrative structures and identifying logic gaps. The process begins with the model receiving a sentence s, a list of characters $C = \{c_1, c_2, \ldots, c_m\}$, and the story's context S for reference.

Guided by prompt engineering (details in Appendix E), MLD-EA processes each sentence to identify and represent the actions performed by the characters as flowing:

For each character c in the characters list C, the model outputs an action in the following format: `<c>Action(Target, Object)</c>`, where c represents the character acting; `Action` denotes the action the character performs; `Target` is the target of the action (who or what the action is directed towards); `Object` specifies any object associated with the action (if applicable). If a character c does not perform any action in the sentence s, the model needs to output: `<c>None</c>`.

4.2 Emotion Classification

The emotion classification module in the MLD-EA categorizes characters' emotions based on given sentences. This classification is based on eight basic emotion types from Plutchik's model (Plutchik, 2001) —*joy, trust, fear, surprise, sadness, disgust, anger, and anticipation*—plus an additional "*none*" category for cases where no emotion is detected.

Before classifying emotions, the model first checks whether each character c in the list $C = \{c_1, c_2, \ldots, c_m\}$ is affected by the events described in each sentence s. If the model determines that a character c is not affected, the emotion for that character is classified as `none`. In addition to the emotion classification, the model also outputs whether or not each character is affected by the sentence.

The model's output for each character c includes the result of the '*affected*' and the emotion classification in `<c>(Affected,` $e(c, s)$`)</c>`, where

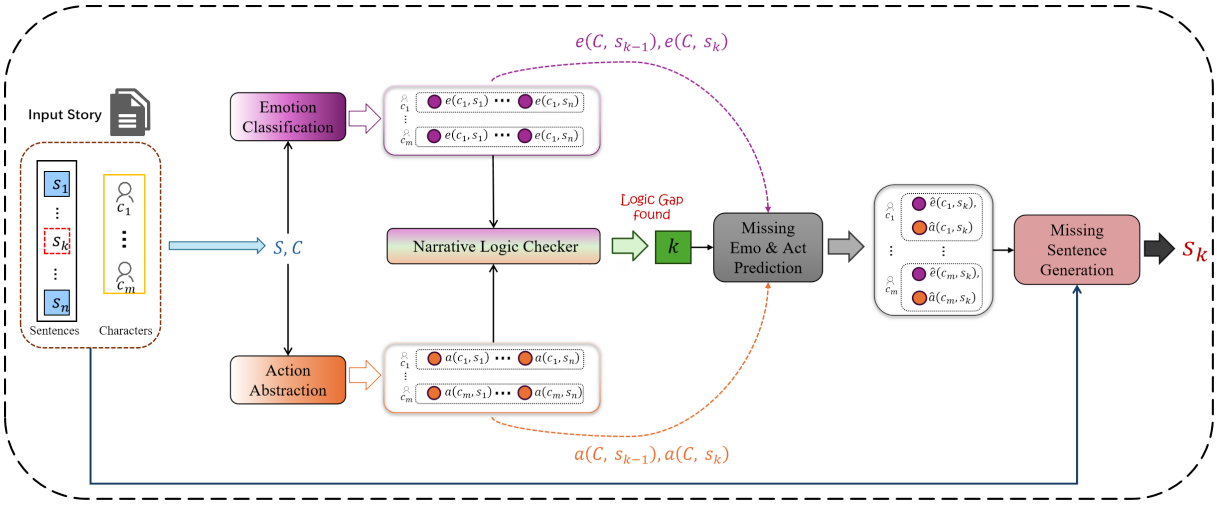

Figure 2: MLD-EA model overview. Each Input Story contains n sentences and m characters, which have a missing sentence s_k before index k. $e(c, s)$ and $a(c, s)$ denote the character's emotion and action in the sentence, respectively; $\hat{e}$ and $\hat{a}$ denotes the predicted emotion and action.

`Affected` is a boolean value indicating whether the character c is affected by any event in the sentence s and $e(c, s)$ represents the emotion associated with the character in sentence s, where $e \in \{$`joy`, `trust`, `fear`, `surprise`, `sadness`, `disgust`, `anger`, `anticipation`, `none`$\}$.

4.3 Narrative Logic Checker By Characters' Emotion and Action

The narrative logic checker component focuses on detecting potential gaps in the narrative by analyzing the relationship between characters' actions and emotions. This process is grounded in the outputs from the previous modules: action abstraction and emotion classification. The prediction is based on detecting disruptions or inconsistencies in each character's expected flow of actions and emotions.

Several key principles in behavior research (Cannon, 1927; Zhu and Thagard, 2002) guide this process: 1), emotions often drive actions. 2), actions can influence subsequent emotions. 3), and some actions directly reflect the character's current emotional state, and vice versa.

MLD-EA then predicts the missing sentence index k, which is determined by evaluating the continuity and logical consistency of the sequences with the interaction of characters' actions and emotions:

$$(E, A) = \sum_{s \in S, c \in C} [e(c, s), a(c, s)], \quad (2)$$

$$k = \text{Inf}_{Index} [(S \oplus (E, A)), C], \quad (3)$$

where Inf_{Index} represent the model inference of missing sentence index prediction. A significant deviation from expected values suggests a missing sentence, and k identifies the position where this sentence should be inserted.

4.4 Action/ Emotion prediction and sentence generation

Following the identification of the missing sentence index by analyzing characters' actions and emotions, the next crucial step in the MLD-EA framework is to predict the actions and emotions of the missing sentence and subsequently generate the sentence. This process is essential to ensure the narrative remains coherent and logically consistent. The focus here is on the immediate context surrounding the predicted index. By examining the sequences of preceding actions and emotions and succeeding actions and emotions, the model estimates the most coherent actions $\hat{a}(c, s_k)$ and emotions $\hat{e}(c, s_k)$ for the missing sentence s_k:

$$[\hat{a}, \hat{e}] = \text{Inf}_{eap} [(a(c, s_{k-1}), e(c, s_{k-1})), (a(c, s_k), e(c, s_k))], \quad (4)$$

where Inf_{eap} means the model inference of emotion and action prediction for the missing sentence. Once these predictions are made, the model generates a sentence to fill the identified gap:

$$s_k = \text{Inf}_{gen}(S, C, k, (\hat{a}, \hat{e})), \quad (5)$$

where Inf_{gen} is a zero-shot inferring. This generated sentence encapsulates the character's possible emotion and action, thereby maintaining the story's coherence and flow and completing the narrative.

5 Experiment

5.1 Data

We use the Story Commonsense dataset for our task, which contains 4853 five-sentence stories with labeled emotions and motivation for characters (Rashkin et al., 2018). We only take the stories with labeled emotion because the labeled motivations are based on Maslow's needs (Maslow, 1943) and Reiss' motives (Reiss, 2004) theory, which are focused on the deeper motivation, not actual actions. By excluding motivations, which are more abstract and theoretical in nature, the analysis remains more grounded in observable narrative events, avoiding complexities that may not directly influence the characters' visible actions. This also ensures that the model can better focus on the emotional states that drive the characters' responses, making it easier to align predictions with surface-level events in the story. We then divided the data into 8:1:1 for training, validation, and testing.

To follow the task of emotion classification in section 4.2 and the task of narrative logic checker in section 4.3, we consolidate the characters' emotions into a single tag by selecting the one with the highest confidence, as determined by three annotators in the original dataset. The details of choosing the missing sentence are in Appendix A.

5.2 Selected Baselines

We compare MLD-EA with the following baselines trained by different strategies and datasets:

Llama3-8B-Instruct (AI@Meta, 2024): Meta's Llama3-8B-Instruct model is a cutting-edge LLM renowned for its exceptional ability to follow instructions meticulously. It is adept at crafting stories that are not only imaginative but also adhere to logical structures and factual integrity.

Gemma2-2B-it (Team, 2024): Gemma2-2B-it is a nimble and efficient model that packs a punch regarding text generation capabilities from Google. Despite its smaller size than some of its peers, it demonstrates remarkable skill in spinning engaging stories that captivate audiences.

Gemma2-9B-it (Team, 2024): Gemma2-9B-it is a larger version of Gemma2-2B-it. With a more vast dataset and bigger model size, it generates intricate and vivid stories rich in detail and depth.

We selected these particular models as baselines for several key reasons: 1) To the best of our knowledge, no prior research has focused on identifying logical gaps or inconsistencies at the sentence level within stories. This novel focus makes it difficult to directly compare our approach to existing studies. 2) While previous works on story generation have primarily relied on pre-trained models such as BERT and GPT-2 (Wang et al., 2022; Paul and Frank, 2021), our study specifically aims to evaluate the capabilities of newer LLMs. The baselines we selected models are all modern LLMs known for their advanced narrative understanding abilities. These models are particularly well-suited for complex tasks related to narrative. 3) We intentionally included models of different sizes and architectures to provide a comprehensive evaluation. This range allows us to compare varying complex models to understand how size and dataset diversity impact logical story generation.

5.3 Implement Setups

MLD-EA is built based on Llama3-8B-Instruct (AI@Meta, 2024) using the Huggingface's libraries[1] (Wolf, 2019) and use Llama-Factory (Zheng et al., 2024) for supervised fine-tuning (Gunel et al., 2020). We use LoRA (Hu et al., 2021) to fine-tune our model. Please see Appendix B for hyper-parameters details and Appendix E for prompts technics we used and prompt templates.

We compute the micro-averaged result of all baselines by the same zero-shot (Wei et al., 2021), one-shot, and few-shot (Brown, 2020) prompts with original input labels from the dataset. All experiments run on two RTX 4090 24GB GPUs.

5.4 Evaluation Metrics

We use the following metrics to evaluate MLD-EA performance on the different sub-tasks:

(1) Both **BLEU-1,2** (Papineni et al., 2002) and **ROUGE-L** (Lin, 2004) are used for evaluating the action abstraction task.

(2) We compute the micro-average **Precision**, **Recall**, and **F1** score for each tag to show the accuracy of emotion classification.

(3) The micro-average **Precision**, **Recall**, and **F1** score are also applied to evaluate the accuracy of the narrative logic checker on each candidate place.

(4) For final generation task based on predicted emotions and actions, we use **BLEU-1,2,4**, **ROUGE-1,2,L** and **BERTScore**[2] (Zhang et al.,

[1] https://huggingface.co/docs

[2] The BERTScore evaluation model is from Hugging Face: https://huggingface.co/spaces/evaluate-metric/bertscore. We used the missing sentence from the original story as the reference and the model-generated sentence as predictions to compute their similarity.

2019) to measure the similarity of candidate sentences and reference sentences. Furthermore, a **Valence-Arousal-Dominance (VAD)** model (Warriner et al., 2013) is used in psychology to describe and measure human emotions. These three dimensions are often used to provide a more comprehensive understanding of emotional states, as they capture different aspects of how emotions are experienced and expressed. We use a developed VAD model (Plisiecki and Sobieszek, 2024) to model the gap between candidate sentences and reference sentences. Also, Plisiecki and Sobieszek (2024) add Age of Acquisition (AoA) and Concreteness as important features in their VAD. AoA refers to the age at which a person learns a particular word or concept. Concreteness measures how tangible or perceptible through the senses a word is.

	BLEU-1	BLEU-2	ROUGE-L
T2Act2T	40.94	34.82	53.67

Table 1: Result of Action Abstraction

Model	P	R	F1
NPN (Rashkin et al., 2018)	24.33	40.10	30.29
Llama3-8B-Instruct	36.20	35.51	35.23
Ours	43.55	42.68	42.98
Ours-*affected*	**48.51**	**50.33**	**49.03**

Table 2: Result of Emotion Classification. The best performance is highlighted in bold, where 'Ours-*affected*' means we consider the '*affected*' features during classification, and the affective features denote whether a character is influenced by any emotion.

6 Results and Analysis

6.1 Action Abstraction and Emotion Classification

The action abstraction has summarized the key concept from the original sentence as open-text, so we evaluate it by creating a simple process called '*Text to Action to Text (T2Act2T)*'. *T2Act2T* takes the abstracted actions at first, and then it generates a new sentence only based on the abstracted actions. In the end, we compare the original sentence with the generated sentence to see how much information remained during the MLD-EA's action abstraction module. Table 1 shows the result between original sentence and new sentence, which illustrates the degree of information kept by our method.

We give results for emotion classification in Table 2. Our model performs best compared to Llama3-8B-Instruct baseline and a developed NPN model (Bosselut et al., 2017) in ROCStories dataset (Rashkin et al., 2018). After fine-tuning, the model archives a significant improvement in emotion classification. Also, when incorporating the '*affected*' feature to detect whether any emotion influences a character, our model attains an impressive F1 score of 88.51 on evaluating the accuracy of '*affected*', respectively. Our findings suggest that including features that account for emotional impact can dramatically improve classification performance, which has implications for various applications in natural language processing.

Furthermore, the partition relationship between the number of labels and their classified accuracy is in Figure 3. Classes with fewer instances show lower accuracy, indicating a need for better representation or enhanced feature engineering to improve performance across less frequent emotions.

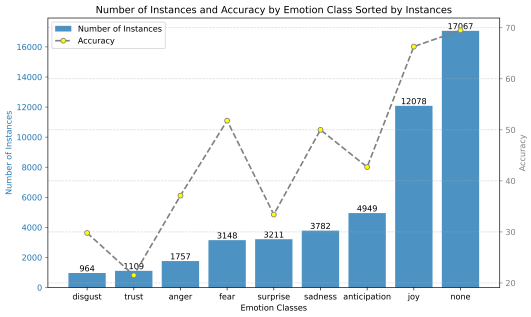

Figure 3: Emotion classification. This figure illustrates the relationship between the number of instances for each emotion class and the corresponding classification accuracy. Classes with more instances, such as 'joy,' exhibit higher classification accuracy compared to less frequent classes like 'disgust' and 'trust,' reflecting the potential influence of data imbalance on performance.

6.2 Narrative Logic Checker

Table 3 presents the results of the narrative logic checker on predicting the index of missing part, which evaluates our model against various baselines both with and without incorporating actions and emotions[3]. MLD-EA model consistently outperforms all baselines across different sentence insertion points. Notably, including actions and emotions significantly improves the micro-averaged F1 scores for all baseline models. Specifically, when the story is complete ($k = -1$), there is a marked

[3]The full results of baselines are shown in Appendix C

Model	k=-1			k=2			k=3			k=4			Avg		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Without **EA**															
Llama3-8B-Instruct	0.00	0.00	0.00	17.90	64.43	24.80	89.72	29.47	44.29	14.04	57.82	19.47	30.41	37.93	22.14
Gemma2-2B-it	0.00	0.00	0.00	2.01	38.46	3.81	86.47	28.06	42.31	7.68	36.24	12.42	24.04	25.69	14.64
Gemma2-9B-it	22.44	21.73	21.79	23.94	83.03	29.37	81.45	34.93	48.28	31.58	66.97	41.23	39.85	51.66	35.17
With **EA***															
Llama3-8B-Instruct*	0.64	11.11	1.21	32.44	48.25	27.86	56.39	32.11	40.75	39.47	53.98	35.27	32.24	36.11	26.27
Gemma2-2B-it*	15.38	10.74	9.53	60.18	34.29	43.67	29.32	27.67	26.88	6.36	60.13	10.56	27.81	33.21	22.66
Gemma2-9B-it*	29.49	37.83	28.62	24.61	54.55	30.51	73.18	34.58	46.36	38.60	67.50	48.09	41.47	48.62	38.39
MLD-EA (Ours)	93.44	100.00	96.61	73.08	74.03	73.55	82.93	56.67	67.33	54.79	85.10	66.67	**76.06**	**78.95**	**76.04**

Table 3: Result of Narrative Logic Checker on predicting missing sentence position. The best performance on average is highlighted in bold. k**=-1**: The input story is completed; k**=2,3,4**: The missing one should be inserted before index[2,3,4], where story's index starts at 1; **Avg**: the Micro-average score of all index's F1 score; Without **EA**: prediction without involving emotions and actions. With **EA** and *****: prediction involving emotions and actions.

Model	BLEU			ROUGE			BERTScore		
	-1	-2	-4	-1	-2	-L	P	R	F1
Without **EA**									
Llama3-8B-Instruct	33.77	4.05	0.28	25.98	5.56	22.27	77.66	79.03	78.33
Gemma2-2B-it	30.31	2.88	0.15	23.36	3.92	20.19	76.91	78.36	77.67
Gemma2-9B-it	33.82	3.38	0.14	24.42	4.03	20.84	77.76	78.94	78.34
With **EA***									
Llama3-8B-Instruct*	36.29	5.83	0.54	28.18	7.18	23.99	77.59	78.98	78.27
Llama3-8B-Instruct$^{\mp}$*	43.68	12.15	**2.67**	34.77	14.23	31.13	77.91	79.34	**78.61**
Gemma2-2B-it*	33.74	6.35	1.84	28.30	8.66	25.52	76.59	78.52	77.54
Gemma2-2B-it$^{\mp}$*	35.98	7.80	2.42	31.03	11.37	27.35	76.54	78.25	77.37
Gemma2-9B-it*	37.27	4.99	0.35	27.08	5.78	23.65	**77.92**	78.90	78.40
Gemma2-9B-it$^{\mp}$*	40.14	7.83	0.91	30.56	9.01	26.82	77.87	**79.21**	78.53
Pre-training Models									
COINS† (Paul and Frank, 2021)	22.82	10.52	-	-	-	19.4	-	-	-
CHAE† (Wang et al., 2022)	32.04	**15.89**	-	-	-	-	-	-	-
COG-BART† (Xie et al., 2022)	24.51	2.26	0.16	18.71	3.11	17.24	-	-	-
MLD-EA (Ours)	**43.92**	12.17	2.29	**35.51**	**14.48**	**31.41**	76.34	77.84	77.08

Table 4: Result of Missing Sentence Generation. The best performance is highlighted in bold. **EA** and *****: Emotions and Actions involved; $\mp$: Input with the action-emotion prediction of the missing sentence. †: The results are taken from the highest scores from their research output.

improvement in F1 scores for each baseline model, underscoring the critical role that action and emotion play in maintaining story logic.

The superior performance of our MLD-EA model highlights its advanced capability to accurately predict the missing sentence in a narrative. This suggests that the model's ability to consider emotional and action-related cues is essential for enhancing the logical coherence of stories. These findings emphasize the importance of incorporating nuanced narrative elements, such as emotions and actions, in developing more sophisticated and reliable models for story generation.

6.3 Sentence Generation

We also compare our model with the baselines for the Generation task, which considers the different situations. Also, we add the influence of action-emotion prediction on generation task. The results, as shown in Table 4, demonstrate that our MLD-EA model, particularly when incorporating predicted actions and emotions, achieves competitive performance across multiple metrics. Notably, our model with the action-emotion prediction achieves the highest scores in several key areas: BLEU-1, BLEU-2, and all ROUGE. Moreover, we notice that BLEU-4 rises dramatically after involving emotions and actions for the Gemma2-2B-it model. This means this method may be more suitable for small-size LLMs on generation tasks with consistency and coherency. We also compare this with previous studies in story plot generation, which are done by pre-training models. Obviously, the LLMs-based results achieve impressive improvement in generation tasks.

Incorporating actions and emotions into the generation process significantly enhances the model's performance, as evidenced by the notable improvement in BLEU and ROUGE scores across all baselines. However, the difference in BERTScore is slight. Overall, the baselines involved in emotion and action while adding action-emotion prediction still outperform the fundamental baselines.

We also use VAD to measure the deviation from the original sentence with model generation. Table

5 concludes that both LLMs can make the generated sentence closer to an original sentence in emotional dimensions after introducing emotions and actions. This improvement indicates that incorporating emotional and action cues enhances the logical consistency of the narrative and ensures that the generated content aligns more closely with the emotional tone of the original text, making the output more authentic and contextually appropriate.

Model	V	A	D	MEAN	AoA	Con
Without **EA**						
Llama3-8B-Instruct	0.160	0.095	0.122	0.126	0.068	0.140
Gemma2-2B-it	0.176	0.092	0.129	0.133	0.069	0.160
Gemma2-9B-it	0.154	0.093	0.113	0.120	0.070	0.153
With **EA***						
Llama3-8B-Instruct*	0.157	0.092	0.123	0.124	0.064	0.143
Gemma2-2B-it*	0.165	0.092	0.111	0.123	**0.063**	0.148
Gemma2-9B-it*	0.143	0.095	**0.110**	**0.116**	0.066	0.154
MLD-EA (Ours)	**0.142**	**0.092**	0.116	0.117	0.065	**0.137**

Table 5: VAD: deviation between original sentence and generated sentence. The closest result is highlighted in bold; **V**: Valence; **A**: Arousal; **D**: Dominance; **MEAN**: mean values of VAD; **AoA**: Age of Acquisition; **Con**: Concreteness; All values range from 0 to 1.

Model	P	R	F1
MLD-EA	81.09	81.19	80.89
w/o ae	77.20	77.24	77.12 ↓ 2.97
w/o a	69.28	69.23	69.16 ↓ 10.93
w/o e	66.47	66.84	66.64 ↓ 13.45

Table 6: Ablation Study of Narrative Logic Checker on predicting missing sentence position with conditional training. **w/o ae**: without actions and emotions; **w/o a**: without actions, emotions only; **w/o e**: without emotions, actions only.

6.4 Ablation Study

MLD-EA's primary task is to find the logic gap by providing characters' emotions and actions. So, we focus on how actions and emotions affect the model's performance in ablation studies. As shown in Table 6[4], after we remove actions (**w/o a**) and emotions (**w/o e**), the accuracy of prediction drops out a lot, which decreases by more than 10 on micro-average F1 score. Surprisingly, we notice if we do not add actions and emotions as input (**w/o ae**), the prediction result is not bad. It may be because the default considering features is unrelated to characters' behaviors; it may be more related to the inherent connection between sentences.

[4]Original emotions labels fine-tune all model here

Input Story:	s_0: "*Gary was looking for a new laptop.*" s_1: "*Gary needed the laptop to be light, fast, and efficient.*" s_2: "*After researching, Gary decided he would purchase a Mac Air.*" s_3: "*Gary was not pleased with the operation of his new laptop at all.*"
Character:	Gary
Actions:	LookingFor(a new laptop) → Needed(laptop) → Decided(Purchase a Mac Air) → NotPleased(laptop)
Emotion:	anticipation → anticipation → joy → anger
Missing Sentence Index:	Insert before sentence **3**.
Missing Sentence:	"*After purchasing, Gary quickly realized he made the wrong decision.*"
Without **EA**	
Llama3-8B-Instruct	Insert before: 2 Sentence: "*He had heard great reviews about the Mac Air's performance and durability.*"
Gemma2-2B-it	Insert before: 2 Sentence: "*Despite its sleek design and powerful processor, the Mac Air seemed to be plagued by frequent crashes and slowdowns.*"
Gemma2-9B-it	Insert before: 3 Sentence: "*Despite its sleek design and positive reviews, the Mac Air proved to be frustratingly slow.*"
With **EA**	
Llama3-8B-Instruct	Insert before: 3 Sentence: "*Gary was frustrated with the slow performance and poor battery life of his new laptop.*"
Gemma2-2B-it	Insert before: 2 Sentence: "*After reading reviews and comparing specs, Gary felt a surge of excitement at the prospect of finally getting a Mac Air.*"
Gemma2-9B-it	Insert before: 3 Sentence: "*However, the Mac Air proved to be much slower than he had anticipated.*"

Table 7: Example: different models with different rules on missing sentence index prediction and generation.

6.5 Case study

Table 7 shows the result of models in finding the missing one's index and generating the sentence before and after the introduction of action and emotion. Without actions and emotions, most models incorrectly predicted the missing location, generating sentences that did not align with the emotional progression. For example, Llama3-8B-Instruct suggested inserting a sentence before s_2 that did not logically lead to Gary's later frustration.

When actions and emotions were included, model performance improved significantly. Both Llama3-8B-Instruct and Gemma2-9B-it accurately identified the correct index and generated sentences that better reflected the emotional shift from joy to anger, such as "*Gary was frustrated with the slow performance and poor battery life of his new laptop.*". The example of "*However, the Mac Air proved to be much slower than he had anticipated.*" even reflects the previous emotion status, making the sentence more connective to the story's consis-

tent emotions and actions. This case study highlights the importance of action-emotion modeling in enhancing the accuracy and coherence of narrative generation, leading to more logically consistent and emotionally resonant outputs.

7 Conclusion

In this work, we introduced the MLD-EA model, a novel approach that leads LLMs to address gaps in narrative logic by integrating actions and emotions. MLD-EA extracts the actions and emotions of the characters in the input story and guides LLMs to find logical loopholes in the narrative by following the rules of interaction between actions and emotions. After getting the position where the missing part should be inserted, it combines the character behaviors and emotions in the context of the missing position to predict the possible character actions and emotions and complete the missing plot. The experimental results demonstrate that MLD-EA significantly improves narrative coherence and emotional alignment compared to existing models, highlighting its effectiveness in story logic detection and generation. By focusing on the interplay between actions and emotions, we have shown that maintaining logical consistency is crucial for producing believable and emotionally resonant narratives. This work advances the field of checking story logic and showcases the potential of LLMs as powerful tools for ensuring narrative cohesion.

Limitations

First, the model has only been tested on short, five-sentence stories and has yet to be evaluated on longer, more complex narratives. This may limit its generalizability to extended storytelling contexts. Second, the model's performance heavily relies on the quality of the original emotion labels and action abstractions. Any inaccuracies in these inputs could negatively affect the model's ability to generate coherent and logically consistent narratives. Future work should address these limitations by testing the model on longer stories and improving the robustness of emotion and action extraction.

Acknowledgments

This work is supported by the Alan Turning Institute/DSO grant: Improving multimodality misinformation detection with affective analysis. Yunfei Long, and Jinming Zhang acknowledge the financial support of the School of Computer Science and Electrical Engineering, University of Essex.

References

AI@Meta. 2024. Llama 3 model card.

Alberto Alvarez. 2023. Chatgpt as a narrative structure interpreter. In *International Conference on Interactive Digital Storytelling*, pages 113–121. Springer.

Ishitva Awasthi, Kuntal Gupta, Prabjot Singh Bhogal, Sahejpreet Singh Anand, and Piyush Kumar Soni. 2021. Natural language processing (nlp) based text summarization-a survey. In *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, pages 1310–1317. IEEE.

Albert Bandura. 1977. Social learning theory. *Prentice-Hall google schola*, 2:101–123.

Antoine Bosselut, Omer Levy, Ari Holtzman, Corin Ennis, Dieter Fox, and Yejin Choi. 2017. Simulating action dynamics with neural process networks. *arXiv preprint arXiv:1711.05313*.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Walter B Cannon. 1927. The james-lange theory of emotions: A critical examination and an alternative theory. *The American journal of psychology*, 39(1/4):106–124.

Charles S Carver, Steven K Sutton, and Michael F Scheier. 2000. Action, emotion, and personality: Emerging conceptual integration. *Personality and social psychology bulletin*, 26(6):741–751.

Gregory Currie and Jon Jureidini. 2004. Narrative and coherence. *Mind & language*, 19(4):409–427.

Sabine A Döring. 2003. Explaining action by emotion. *The Philosophical Quarterly*, 53(211):214–230.

Nancy Eisenberg. 2014. *Altruistic emotion, cognition, and behavior (PLE: Emotion)*. Psychology Press.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898.

Rachelyn Farrell and Stephen G Ware. 2024. Planning stories neurally. *Authorea Preprints*.

Jian Guan, Fei Huang, Zhihao Zhao, Xiaoyan Zhu, and Minlie Huang. 2020. A knowledge-enhanced pretraining model for commonsense story generation. *Transactions of the Association for Computational Linguistics*, 8:93–108.

Jian Guan, Xiaoxi Mao, Changjie Fan, Zitao Liu, Wenbiao Ding, and Minlie Huang. 2021. Long text generation by modeling sentence-level and discourse-level

coherence. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6379–6393.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. Supervised contrastive learning for pre-trained language model fine-tuning. *arXiv preprint arXiv:2011.01403*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Guangming Huang, Yunfei Long, Cunjin Luo, Jiaxing Shen, and Xia Sun. 2024a. Prompting explicit and implicit knowledge for multi-hop question answering based on human reading process. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13179–13189.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024b. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*.

Hanlei Jin, Yang Zhang, Dan Meng, Jun Wang, and Jinghua Tan. 2024. A comprehensive survey on process-oriented automatic text summarization with exploration of llm-based methods. *arXiv preprint arXiv:2403.02901*.

Subbarao Kambhampati, Karthik Valmeekam, Lin Guan, Kaya Stechly, Mudit Verma, Siddhant Bhambri, Lucas Saldyt, and Anil Murthy. 2024. Llms can't plan, but can help planning in llm-modulo frameworks. *arXiv preprint arXiv:2402.01817*.

Robert L Leahy, David A Clark, and DJ Dozois. 2022. Cognitive-behavioral theories. *Gabbard's Textbook of Psychotherapeutic Treatments*, 151.

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.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. 2022. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35:4328–4343.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Qiang Lu, Xia Sun, Yunfei Long, Zhizezhang Gao, Jun Feng, and Tao Sun. 2023. Sentiment analysis: Comprehensive reviews, recent advances, and open challenges. *IEEE Transactions on Neural Networks and Learning Systems*.

Qiang Lu, Xia Sun, Yunfei Long, Xiaodi Zhao, Wang Zou, Jun Feng, and Xuxin Wang. 2025. Multimodal dual perception fusion framework for multimodal affective analysis. *Information Fusion*, 115:102747.

AH Maslow. 1943. A theory of human motivation. *Psychological Review google schola*, 2:21–28.

Raymond J Mooney and Gerald DeJong. 1985. Learning schemata for natural language processing. In *IJCAI*, pages 681–687.

Keith Oatley. 2002. Emotions and the story worlds of fiction. *Narrative impact: Social and cognitive foundations*, 39:69.

OpenAI. 2022. Introducing chatgpt. https://openai.com/index/chatgpt/. Accessed: 2024-08-07.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Zeeshan Patel, Karim El-Refai, Jonathan Pei, and Tianle Li. 2024. Swag: Storytelling with action guidance. *arXiv preprint arXiv:2402.03483*.

Debjit Paul and Anette Frank. 2021. Coins: Dynamically generating contextualized inference rules for narrative story completion. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5086–5099.

Hubert Plisiecki and Adam Sobieszek. 2024. Extrapolation of affective norms using transformer-based neural networks and its application to experimental stimuli selection. *Behavior Research Methods*, 56(5):4716–4731.

Robert Plutchik. 2001. The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American scientist*, 89(4):344–350.

Hannah Rashkin, Antoine Bosselut, Maarten Sap, Kevin Knight, and Yejin Choi. 2018. Modeling naive psychology of characters in simple commonsense stories. *arXiv preprint arXiv:1805.06533*.

Steven Reiss. 2004. Multifaceted nature of intrinsic motivation: The theory of 16 basic desires. *Review of general psychology*, 8(3):179–193.

Gemma Team. 2024. Gemma.

Ashish Vaswani. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Xinpeng Wang, Han Jiang, Zhihua Wei, and Shanlin Zhou. 2022. Chae: Fine-grained controllable story generation with characters, actions and emotions. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6426–6435.

Yichen Wang, Kevin Yang, Xiaoming Liu, and Dan Klein. 2023. Improving pacing in long-form story planning. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Amy Beth Warriner, Victor Kuperman, and Marc Brysbaert. 2013. Norms of valence, arousal, and dominance for 13,915 english lemmas. *Behavior research methods*, 45:1191–1207.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

T Wolf. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Kaige Xie and Mark Riedl. 2024. Creating suspenseful stories: Iterative planning with large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2391–2407.

Yuqiang Xie, Yue Hu, Wei Peng, Guanqun Bi, and Luxi Xing. 2022. Comma: Modeling relationship among motivations, emotions and actions in language-based human activities. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 163–177.

Zhuohan Xie, Trevor Cohn, and Jey Han Lau. 2023. The next chapter: A study of large language models in storytelling. In *Proceedings of the 16th International Natural Language Generation Conference*, pages 323–351.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Qingqing Zhao, Yuhan Xia, Yunfei Long, Ge Xu, and Jia Wang. 2025. Leveraging sensory knowledge into text-to-text transfer transformer for enhanced emotion analysis. *Information Processing & Management*, 62(1):103876.

Zoie Zhao, Sophie Song, Bridget Duah, Jamie Macbeth, Scott Carter, Monica P Van, Nayeli Suseth Bravo, Matthew Klenk, Kate Sick, and Alexandre LS Filipowicz. 2023. More human than human: Llm-generated narratives outperform human-llm interleaved narratives. In *Proceedings of the 15th Conference on Creativity and Cognition*, pages 368–370.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.

Jing Zhu and Paul Thagard. 2002. Emotion and action. *Philosophical psychology*, 15(1):19–36.

Yuchen Zhuang, Yue Yu, Kuan Wang, Haotian Sun, and Chao Zhang. 2024. Toolqa: A dataset for llm question answering with external tools. *Advances in Neural Information Processing Systems*, 36.

A Chosen Missing Sentence

Given a sequence of emotions attributed to characters in a narrative, we determine where emotional changes are most pronounced. Specifically, we analyze the emotions expressed by each character at different steps, calculate the "distance" between emotions in sentences, and identify the step where the aggregate emotional change across all characters is the greatest. This is crucial for understanding key moments in emotional narratives, potentially highlighting climaxes or critical turning points.

For each character c, at current sentence s_i, we calculate the emotion change value $D(e_{s_i}, e_{s_j}, c)$ for each sentence s_j:

$$D(e_{s_i}, e_{s_j}, c) = \begin{cases} \frac{d(e^c_{s_i}, e^c_{s_j})}{|s_i - s_j|} & \textit{if } e_{s_i,c} \\ & \textit{and } e_{s_j,c} \\ 0 & \textit{otherwise} \end{cases} \quad (6)$$

where $d(x)$ represents the function to compute the distance between e^c_{si} and $e^c_{s_j}$. Then, identify the sentence $s_{i_{max}}$ where emotions change maximized:

$$s_{i_{max}} = \arg\max_{s_i} \sum_{c=c_1}^{c_m} \sum_{s_j=s_1}^{s_n} D(e_{s_i}, e_{s_j}, c), \quad (7)$$

where $i_{\max}$ represents the index in the sequence where the emotions across all characters experience the greatest change. Then we remove this $s_{i_{max}}$ from the original story.

B hyper-parameters Used in MLD-EA

Table 8 shows hyper-parameters of fine-tuning. The generation tasks' hyper-parameters for all models are the same as shown in Table 9.

C Details of Narrative Logic Checker result on predicting missing sentence index

Table 10 shows all results of the narrative logic checker running on baselines with different prompt techniques we used in the experiment.

Parameter name	Value
lora_rank	8
lora_alpha	16
lora_dropout	0.1
lora_target	all
learing rate	$2e-5$
epoches	3

Table 8: hyper-parameters of fine-tuning

Parameter name	Value
torch_dtype	torch.float16
do_sample	True
temperature	0.1
top_p	0.4

Table 9: hyper-parameters of generation

All results of missing sentence index prediction results when involved actions and emotions have increased on average. Especially before involving actions and emotions in inference, they are hard to recognize when the story is completed. However, after we add actions and emotions during inferring, the LLMs can recognize the completed story even with the zero-shot prompt (**Gemma2-9B-it***). These results illustrate that considering the interaction between actions and emotions can extraordinarily improve LLMs' narrative logic checking.

D Error Analysis: Generation results with correct index

One key area for error analysis involves evaluating how well the model predicts the correct index for the missing sentence. Misplacement of the generated sentence can disrupt the logical flow of the narrative. Table 11 shows the generation results when the input of the missing sentence index is correct. In this evaluation, we focused on how predicted action-emotion affects the generation quality of the missing part. So, we will only consider when the index is predicted correctly by the narrative logic checker in relation to the analysis results, which involve emotion and actions.

The results show the importance of when models predict the index of logical loopholes. The change of BLEU and ROUGE remains the same because they are all compared with the reference story. At the sentence level BERTScore measures, the F1 score increases dramatically if the generated sentence is filled in the right place. This highlights the model's ability to produce more contextually appropriate and coherent content when the narrative gap is accurately identified. This underscores the importance of accurate index prediction in generating logically and emotionally consistent stories.

E Prompt Engineering

We started at zero-shot for all the cases, then developed one-shot and few-shots after confirming the zero-shot prompt template. Also, we used Chain-of-Thought as an assistant prompt strategy.

For emotion classification, we begin our approach by deploying a suite of meticulously designed prompts to leverage the MLD-EA's capabilities in emotion classification, guiding the model to accurately discern and categorize the emotional spectrum associated with each character in a given sentence. After establishing a baseline performance using the inherent strengths of LLMs, we refine our MLD-EA model through a process inspired by the baseline. This refinement is achieved using supervised fine-tuning with a custom-tailored prompt that enhances the model's ability to detect and classify emotions more precisely for individual characters. This targeted fine-tuning boosts the model's proficiency, enhancing its analytical and emotional sentiment analysis capabilities.

There are some examples of prompt templates used for baselines on experiments. Table 12 shows the prompt template for action abstraction. We also present the prompt templates for both '*Without **EA***' and '*With **EA***' for the narrative logic checker and generation tasks. The prompt templates of '*Without **EA***' mean the LLMs need to find the logic loopholes and complete the plot only along with the input story, which the zero-shot prompt template for the narrative logic checker is shown in Table 13 and the generation template is in Table 14. The prompt templates of '*With **EA***' means the LLMs have to consider the characters' emotions and actions during those tasks, which the zero-shot prompt template for the narrative logic checker is shown in Table 13 and the generation template is in Table 14. Also, Table 15 shows how we predict the actions and emotions for the missing part. Notably, All the results of '*Without **EA***' are actually how LLMs face those tasks without any further information. Our study considers the interaction of actions and emotions to increase overall performances for LLMs on those tasks.

Actions and Emotions	Model	k=-1			k=2			k=3			k=4			Avg		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Without **EA**	**Llama3-8B-Instruct**															
	zero-shot	0.00	0.00	0.00	10.73	57.14	18.08	95.49	27.97	43.27	0.00	0.00	0.00	26.56	21.28	15.34
	one-shot	0.00	0.00	0.00	35.57	53.53	42.74	78.95	27.70	41.02	2.63	50.00	5.00	29.29	32.81	22.19
	few-shot	0.00	0.00	0.00	12.75	73.08	21.71	90.98	32.27	47.64	36.18	67.90	47.21	34.98	43.31	29.14
	Gemma2-2B-it															
	zero-shot	0.00	0.00	0.00	2.68	57.14	5.13	98.50	28.42	44.10	5.26	44.44	9.41	26.61	32.50	14.66
	one-shot	0.00	0.00	0.00	1.34	15.38	2.47	76.69	26.98	39.92	0.00	0.00	0.00	19.51	10.59	10.60
	few-shot	0.00	0.00	0.00	2.01	42.86	3.85	84.21	28.79	42.91	17.76	64.29	27.84	26.00	33.98	18.65
	Gemma2-9B-it															
	zero-shot	0.00	0.00	0.00	3.36	1.00	6.49	95.49	29.74	45.36	23.68	67.92	35.12	30.62	49.42	21.74
	one-shot	32.69	24.29	27.87	55.03	62.12	58.36	68.42	38.89	49.59	23.68	73.47	35.82	44.96	49.69	42.91
	few-shot	34.62	40.91	37.50	13.42	86.96	23.26	80.45	36.15	49.88	47.37	59.50	52.75	43.96	55.88	40.85
With **EA***	**Llama3-8B-Instruct***															
	zero-shot	0.00	0.00	0.00	77.86	39.46	52.37	36.09	25.40	29.81	1.32	66.67	2.58	28.81	32.88	21.19
	one-shot	0.00	0.00	0.00	14.09	43.75	21.32	57.89	33.19	42.19	52.63	41.24	46.24	31.16	29.54	27.44
	few-shot	1.92	33.33	3.64	5.37	61.54	9.88	75.19	37.74	50.25	64.47	51.04	56.98	36.74	45.91	30.19
	Gemma2-2B-it*															
	zero-shot	0.00	0.00	0.00	65.77	35.90	46.45	36.84	23.90	28.99	3.29	71.43	6.29	26.48	32.81	20.43
	one-shot	38.46	11.17	17.32	46.98	28.57	35.53	12.03	27.59	16.75	0.66	50.00	1.30	24.53	29.33	17.73
	few-shot	7.69	21.05	11.27	67.79	38.40	49.03	39.10	31.52	34.90	15.13	58.97	24.08	32.43	37.49	29.82
	Gemma2-9B-it*															
	zero-shot	1.92	33.33	3.64	12.75	57.58	20.88	90.23	30.77	45.89	26.32	66.67	37.74	32.80	47.09	27.04
	one-shot	38.46	44.44	41.24	48.32	51.80	50.00	61.65	37.10	46.33	36.18	70.51	47.83	46.16	50.96	46.35
	few-shot	48.08	35.71	40.98	12.75	54.29	20.65	67.67	35.86	46.88	53.29	65.32	58.70	45.45	47.79	41.80

Table 10: Details of Narrative Logic Checker result on predicting missing sentence index. **-1**: The input story is completed; **2,3,4**: The missing sentence should be inserted before index[2,3,4], where story's index starts at 1; **Avg**: the Micro-average score of all index's F1 score; Without **EA**: prediction without involving emotions and actions. With **EA** and *: prediction involving emotions and actions.

Model	BLEU			ROUGE			BERTScore		
	-1	-2	-4	-1	-2	-L	P	R	F1
Llama3-8B-Instruct	43.68	12.15	2.67	34.77	14.23	31.13	77.91	79.34	78.61
Llama3-8B-Instruct*	44.26	12.56	2.93	35.08	14.58	31.46	87.35	88.91	88.11
Gemma2-2B-it	35.98	7.80	2.42	31.03	11.37	27.35	76.54	78.25	77.37
Gemma2-2B-it*	36.42	6.96	1.55	30.34	10.15	27.08	80.81	81.95	81.04
Gemma2-9B-it	40.14	7.83	0.91	30.56	9.01	26.82	77.87	79.21	78.53
Gemma2-9B-it*	40.63	8.04	0.84	30.50	9.31	26.88	87.01	88.53	87.75

Table 11: Generation results with correct index. The model with *: Input with the correct prediction of the missing sentence index. All results are based on the correct missing sentence index as input.

Instruction:
You are an AI designed to abstract and categorize actions from given sentences. You will receive a sentence with a list of characters(The characters may or may not appear in the sentence but appear in the completed story; some Pronouns like He, she, etc. mean one of the characters provided in Input. Do not care about those characters who are not provided).
Your task is to identify any actions these characters perform and abstract them from the sentence in a specific format.
The whole story of this sentence will be provided before the sentence to help you do the mentioned detection.
Format for abstraction:
For each character, specify the action they performed and the target of the action (if any) in the form <Character>Action(Target, ActionObject)</Character>.
(
Character: The character performing the action (i.e. Lucy, I, Lucy's mom, etc.).
Action: The action performed by the character (i.e. Love, Loved, Loves, See, Saw, Attack, Attacks, Attacked, Move, Moves, Moved, Move to, Come, Came, etc.).
Target: The target of the action (who or what the action is directed towards) (i.e. A Love B -> <A>Love(B)</A>).
ActionObject: The specific object related to the action (if any) (i.e. A give b an apply -> <a>Give(b, an apply)).
)
If a character does not perform any action, output <Character>None</Character>.

Table 12: Prompt template: Action Abstraction

Without **EA**

Instruction:
You are an AI assistant designed to analyze and evaluate user inputs for completeness and coherence. Your primary task is to determine whether the provided sequence of sentences is missing a sentence. If you think a sentence is missing, identify where the missing one should be inserted, i.e. if a sentence is missing between sentence 1 and sentence 2, the result should be inserted at index 2; otherwise, if you think no sentence is missing here, just output -1.
UserInput will provide a story with several sentences;

Note that the first and last sentences are constantly provided at the story's start and end; they should not be considered missing. Please find out the index of where the missing one should be inserted before which sentence. Only give the final output, and in this format: Insert before sentence [**i**].

With **EA**

Instruction:
You are an AI assistant designed to analyze and evaluate user inputs for completeness and coherence. Your primary task is to determine whether the provided sequence of sentences is missing a sentence. If you think a sentence is missing, identify where the missing one should be inserted, i.e. if a sentence is missing between sentence 1 and sentence 2, the result should be inserted at index 2; otherwise, if you think no sentence is missing here, just output -1.
UserInput will provide a story with several sentences; characters' actions and emotions in sentences are shown after each sentence.

Consider the following Rules while analyzing:
Rules:
- Emotion affects Action: Actions are often taken because of an emotion.
- Action affects Emotion: Emotions can change due to actions taken.
- Emotion and Action at the same time: Some actions demonstrate current emotions.
- Consider the relationship between the emotions and actions of each character linked between sentences to identify any missing sentence.
- Analyze each character's action and emotion chain to find the missing parts.

Use the given rules and provided data to ensure a logical flow of events and completeness in the narrative. Note that the first and last sentences are constantly provided at the story's start and end; they should not be considered missing. Please find out the index of where the missing one should be inserted before which sentence. Only give the final output, and in this format: Insert before sentence [**i**].

Table 13: Prompt template: Narrative Logic Checker

Without **EA**:

Instruction:
You are an AI assistant (Master in story writing) designed to help users analyze, evaluate, and complete stories by checking their completeness and coherence.
Generate a sentence to fill a gap in a narrative based on the surrounding context, ensuring the story remains coherent and complete.

Generate the Missing Sentence:
–Create a sentence that naturally fits into the narrative at the specified index.
–Ensure the new sentence connects logically with the sentences before and after it, maintaining a smooth and coherent flow.
–Match the style and tone of the existing story.

UserInput will provide a story with several sentences, and the index of missing one should be inserted before.

With **EA** but no prediction actions and emotions:

Instruction:
You are an AI assistant (Master in story writing) designed to help users analyze, evaluate, and complete stories by checking their completeness and coherence.
Generate a sentence to fill a gap in a narrative based on the surrounding context, ensuring the story remains coherent and complete.

Generate the Missing Sentence:
–Create a sentence that naturally fits into the narrative at the specified index.
–Ensure the new sentence connects logically with the sentences before and after it, maintaining a smooth and coherent flow.
–Match the style and tone of the existing story.

UserInput will provide a story with several sentences; characters' actions and emotions in sentences are shown after each sentence. The Characters in story and the index of the missing sentence should be inserted before.

With **EA** and prediction actions and emotions:

You are an AI assistant (Master in story writing) designed to help users analyze, evaluate, and complete stories by checking their completeness and coherence.
Generate a sentence to fill a gap in a narrative based on the surrounding context, ensuring the story remains coherent and complete.

Generate the Missing Sentence:
–Create a sentence that naturally fits into the narrative at the specified index.
–Ensure the new sentence connects logically with the sentences before and after it, maintaining a smooth and coherent flow.
–Match the style and tone of the existing story.
–Consider whether the given actions and emotions are reasonable in this situation. Then, generate the sentence.

Notes:
1. The action form looks like this: Action(Target, ActionObject), where
(Action: The action performed by the character (i.e. Love, Loved, Loves, See, Saw, Attack, Attacks, Attacked, Move, Moves, Moved, Move to, Come, Came, etc.).
Target: The target of the action (who or what the action is directed towards) (i.e. A Love B -> A Love(B)).
ActionObject: The specific object related to the action (if any) (i.e. A give b an apply -> A Give(b, an apply)).
)
2. The emotions are ONLY from Plutchik's eight basic emotions (joy, trust, fear, surprise, sadness, disgust, anger, anticipation) for the characters based on their likely emotional state based on the context and characters' actions. If 'none' means the characters do not have a discernible emotion or will not appear at this point.

UserInput will provide a story with several sentences, and the index of missing one should be inserted before. Also, the predicted actions and emotions of characters that may happen in this missing sentence will be given.

Table 14: Prompt template: Sentence Generation

Instruction:
You are an AI assistant (Master in story writing) designed to help users analyze, evaluate and complete stories by checking their completeness and coherence. Especially is good at action analysis and Plutchik's emotion analysis.
Purpose:
Predict the most likely actions and emotions of characters for a sentence that should be inserted before the specified index in a story, ensuring the narrative remains coherent and logically connected. UserInput will provide a story with several sentences, all characters in the story and the index of missing sentences should be inserted. Think it step by step.

Contextual Analysis:
1. Examine the provided story and identify the events leading to the specified index; if the index is -1, no missing sentence needs to be generated here; stop responding and give 'none'.
2. Focus on the actions and emotions of the characters in the story to understand their progression.

Action Prediction:
1. Predict the most likely action that would occur before the specified index. This prediction should be based on strong evidence from the surrounding context and reflect a logical progression in the narrative.
2. The action should be in the open-text format and reflect what the character would logically do next based on previous actions, emotions and the situation.
3. The action form looks like this: Action(Target, ActionObject), where (Action: The action performed by the character (i.e. Love, Loved, Loves, See, Saw, Attack, Attacks, Attacked, Move, Moves, Moved, Move to, Come, Came, etc.).
Target: The target of the action (who or what the action is directed towards) (i.e. A Love B -> A Love(B)).
ActionObject: The specific object related to the action (if any) (i.e. A give b an apply -> A Give(b, an apply)).
)

Emotion Prediction:
1. Assign an emotion ONLY from Plutchik's eight basic emotions (joy, trust, fear, surprise, sadness, disgust, anger, anticipation) to the characters based on their likely emotional state based on the context and characters' actions.
2. If the characters do not have a discernible emotion or will not appear at this point, use 'none'.

Reasoning:
1. Provide the predicted action and emotion for each character(s) that should appear in the missing sentence before the specified index.
2. Ensure that the predicted actions and emotions consistently follow logical flow.

Table 15: Prompt template: Actions and Emotions Prediction